```
DDDDDDDDDDD     EEEEEEEEEEEEEEE  BBBBBBBBBBB      UUU         UUU     GGGGGGGGGGG
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUU         UUU     GGGGGGGGGGG
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUU         UUU     GGGGGGGGGGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG
DDD         DDD EEEEEEEEEEE      BBBBBBBBBBBB     UUU         UUU GGG
DDD         DDD EEEEEEEEEEE      BBBBBBBBBBBB     UUU         UUU GGG
DDD         DDD EEEEEEEEEEE      BBBBBBBBBBBB     UUU         UUU GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG    GGGGGGGGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG    GGGGGGGGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG    GGGGGGGGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG          GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG          GGG
DDD         DDD EEE              BBB         BBB  UUU         UUU GGG          GGG
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
DDDDDDDDDDDD    EEEEEEEEEEEEEEE  BBBBBBBBBBBB     UUUUUUUUUUUUUUU     GGGGGGGGG
```

```
DDDDDDDD    BBBBBBBB      GGGGGGGG    CCCCCCCC  VV      VV  TTTTTTTTTT  DDDDDDDD    XX      XX
DDDDDDDD    BBBBBBBB      GGGGGGGG    CCCCCCCC  VV      VV  TTTTTTTTTT  DDDDDDDD    XX      XX
DD      DD  BB      BB  GG          CC          VV      VV      TT      DD      DD  XX      XX
DD      DD  BB      BB  GG          CC          VV      VV      TT      DD      DD  XX      XX
DD      DD  BB      BB  GG          CC          VV      VV      TT      DD      DD    XX  XX
DD      DD  BB      BB  GG          CC          VV      VV      TT      DD      DD    XX  XX
DD      DD  BBBBBBBB    GG          CC          VV      VV      TT      DD      DD      XX
DD      DD  BBBBBBBB    GG          CC          VV      VV      TT      DD      DD      XX
DD      DD  BB      BB  GG  GGGGGG  CC          VV      VV      TT      DD      DD    XX  XX
DD      DD  BB      BB  GG  GGGGGG  CC          VV      VV      TT      DD      DD    XX  XX
DD      DD  BB      BB  GG      GG  CC            VV  VV        TT      DD      DD  XX      XX      ....
DD      DD  BB      BB  GG      GG  CC            VV  VV        TT      DD      DD  XX      XX      ....
DDDDDDDD    BBBBBBBB      GGGGGG    CCCCCCCC        VV          TT      DDDDDDDD    XX      XX      ....
DDDDDDDD    BBBBBBBB      GGGGGG    CCCCCCCC        VV          TT      DDDDDDDD    XX      XX      ....
```

```
LL              IIIIII      SSSSSSSS
LL              IIIIII      SSSSSSSS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II      SS
LL                II        SSSSSS
LL                II        SSSSSS
LL                II              SS
LL                II              SS
LL                II              SS
LL                II              SS
LLLLLLLLLL      IIIIII    SSSSSSSS
LLLLLLLLLL      IIIIII    SSSSSSSS
```

```
;     1          0001  0 MODULE DBGCVTDX (IDENT = 'V04-000') =
;     2          0002  0
;     3          0003  1 BEGIN
;     4          0004  1 !
;     5          0005  1 !**************************************************************************
;     6          0006  1 !*                                                                        *
;     7          0007  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                              *
;     8          0008  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                *
;     9          0009  1 !*  ALL RIGHTS RESERVED.                                                  *
;    10          0010  1 !*                                                                        *
;    11          0011  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED  *
;    12          0012  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE  *
;    13          0013  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
;    14          0014  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
;    15          0015  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
;    16          0016  1 !*  TRANSFERRED.                                                           *
;    17          0017  1 !*                                                                        *
;    18          0018  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
;    19          0019  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
;    20          0020  1 !*  CORPORATION.                                                           *
;    21          0021  1 !*                                                                        *
;    22          0022  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS  *
;    23          0023  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                *
;    24          0024  1 !*                                                                        *
;    25          0025  1 !*                                                                        *
;    26          0026  1 !**************************************************************************
;    27          0027  1 !
;    28          0028  1 !  WRITTEN BY
;    29          0029  1 !       Farokh Morshed          01-09-1981
;    30          0030  1 !
;    31          0031  1 !  MODIFIED BY:
;    32          0032  1 !       * These modifications are to LIB$$FIND_CVT_PATH, and were done before
;    33          0033  1 !       * debug modifications.
;    34          0034  1 !       *
;    35          0035  1 !       1-001 - Original.  FM1001       01-09-1981
;    36          0036  1 !       1-002 - Put in a check for DSC$W_LENGTH to be 1 when class A, or NCA, and
;    37          0037  1 !               if class NCA stride must be 1.  FM 9-9-81
;    38          0038  1 !       1-003 - Put in a new data type, DSC$K_DTYPE_VT.  FM 1-DEC-81.
;    39          0039  1 !       1-004 - Put in a feature where DST_INFO [D_[EN] can be picked up for
;    40          0040  1 !               LIB$CVT_DX_DX. FM 2-DEC-81.
;    41          0041  1 !
;    42          0042  1 !       * These modifications are to LIB$CVT_DX_DX, and were done before
;    43          0043  1 !       * debug modifications.
;    44          0044  1 !       *
;    45          0045  1 !       1-001 - Original.  FM1001       01-09-1981
;    46          0046  1 !       1-002 - Fix the problem with (SMLINT, LRGINT, DEC) to NBDS having an explicit
;    47          0047  1 !               sign when plus should be implied.  Also [DEC_NBDS] scaled twice,
;    48          0048  1 !               changed it to scale only once.  FM 5-NOV-81.
;    49          0049  1 !       1-003 - Fix the problem with [K_DEC_NBDS].  The length of CLASS_S_DESC was
;    50          0050  1 !               not being reset. FM
;    51          0051  1 !       1-004 - Put in a new data type, DSC$K_DTYPE_VT.  Cleaned up data type B
;    52          0052  1 !               out of NBDS.  FM 1-DEC-81.
;    53          0053  1 !       1-005 - Fix the bug where destination length is not picked up from DST_INFO.
;    54          0054  1 !               FM 2-DEC-81.
;    55          0055  1 !       1-006 - Constants which are addressed by things like PACK_ZERO should be
;    56          0056  1 !               all longwords.
;    57          0057  1 !       1-007 - LIB$_ROPRAND was left out of the exception handler.  FM 8-FEB-82.
```

```
;    58          0058  1 !       1-008 - A couple of missing dots fixed -Q -> G and H.
;    59          0059  1 !
;    60          0060  1 !       * DEBUG modifications start here.
;    61          0061  1 !       *
;    62          0062  1 !       1-001 - Victoria Holt   Sept., 1982
;    63          0063  1 !               Created module DBGCVTDX.  This module includes the two routines
;    64          0064  1 !               FIND_CVT_PATH and DBG$CVT_DX_DX (originally LIB$$FIN_CVT_PATH
;    65          0065  1 !               and LIB$CVT_DX_DX, respectively).  Both routines have been
;    66          0066  1 !               modified to include additional DEBUG and language specific
;    67          0067  1 !               dtypes and classes.
;    68          0068  1 !       1-002 - VJH
;    69          0069  1 !               Added routine DBG$COVER_DX_DX from DBGEVALOP.
;    70          0070  1 !               Modified handler so that it signals errors rather than
;    71          0071  1 !               returning a status code.
;    72          0072  1 !       1-003 - WC3     Jul-83
;    73          0073  1 !               Add support for Absolute Date Time to CVT_DX_DX
;    74          0074  1 !       1-004 - WC3     Jul-83
;    75          0075  1 !               Fix the decimal text to Octaword conversion
;    76          0076  1 !       1-005 - BAB     Dec-83
;    77          0077  1 !               Added support for scaled binary conversions. To and From.
;    78          0078  1 !       1-006 - BAB     Jan-84
;    79          0079  1 !               Fixed the bug where DEP/QUAD I=8000000000000000 does not work.
;    80          0080  1 !               Also DEP/OCTA I=80000000000000000000000000000000.
;    81          0081  1 !
;    82          0082  1
;    83          0083  1 REQUIRE 'SRC$:DBGPROLOG.REQ';
;    84          0217  1
;    85          0218  1 LINKAGE
;    86          0219  1     JSB_R0 = JSB (REGISTER = 0): PRESERVE (0, 1),
;    87          0220  1     JSB_R1 = JSB (REGISTER = 0, REGISTER = 1): PRESERVE (0, 1),
;    88          0221  1     JSB_RETRO_R1 = JSB (REGISTER = 0, REGISTER = 1): PRESERVE (1),
;    89          0222  1     JSB_R2 = JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2): PRESERVE (0, 1),
;    90          0223  1     JSB_R3 = JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3): PRESERVE (0, 1),
;    91          0224  1     JSB_R6 = JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2, REGISTER = 3, REGISTER = 4, REGISTER = 5):
;    92          0225  1     PRESERVE (0, 1),
;    93          0226  1     SCOPYR_JSB_R6 = JSB (REGISTER = 0, REGISTER = 1, REGISTER = 2): NOPRESERVE (2),
;    94          0227  1     SCOPY_JSB_R6 = JSB (REGISTER = 0, REGISTER = 1): NOPRESERVE (2, 3, 4, 5, 6);
;    95          0228  1
;    96          0229  1 FORWARD ROUTINE
;    97          0230  1     DBG$COVER_DX_DX,                        ! Accepts value descriptors; calls DBG$CVT_DX_DX.
;    98          0231  1     COVER_VMSDESC_SETUP,                    ! Set up vms descriptor
;    99          0232  1     DBG$CVT_DX_DX:  NOVALUE,                ! Routine that does any-to-any type conversion.
;   100          0233  1     CVT_HANDLER,                            ! Error handler.
;   101          0234  1     FIND_CVT_PATH;                          ! Routine to find the conversion path
;   102          0235  1                                             ! being done and report any
;   103          0236  1
;   104          0237  1 EXTERNAL ROUTINE
;   105          0238  1     DBG$CVT_ASHP_R1: JSB_R6 NOVALUE,
;   106          0239  1     DBG$CVT_CMPH_R1: JSB_RETRO_R1,
;   107          0240  1     DBG$CVT_CVTDA_R1: JSB_R1 NOVALUE,
;   108          0241  1     DBG$CVT_CVTLB_R1: JSB_R1 NOVALUE,
;   109          0242  1     DBG$CVT_CVTLH_R1: JSB_R1 NOVALUE,
;   110          0243  1     DBG$CVT_CVTLW_R1: JSB_R1 NOVALUE,
;   111          0244  1     DBG$CVT_CVTRDQ_R1: JSB_R1 NOVALUE,
;   112          0245  1     DBG$CVT_CVTHD_R1: JSB_R1 NOVALUE,
;   113          0246  1     DBG$CVT_CVTHF_R1: JSB_R1 NOVALUE,
;   114          0247  1     DBG$CVT_CVTHG_R1: JSB_R1 NOVALUE,
```

```
;  115          0248  1      DBG$CVT_CVTGH_R1: JSB_R1 NOVALUE,
;  116          0249  1      DBG$CVT_CVTRHL_R1: JSB_R1 NOVALUE,
;  117          0250  1      DBG$CVT_CVTRHO_R1: JSB_R1 NOVALUE,
;  118          0251  1      DBG$CVT_CVTRHQ_R1: JSB_R1 NOVALUE,
;  119          0252  1      DBG$CVT_CVTROUD_R1: JSB_R1 NOVALUE,
;  120          0253  1      DBG$CVT_CVTROUH_R1: JSB_R1 NOVALUE,
;  121          0254  1      DBG$CVT_DIVD2_R1: JSB_R1 NOVALUE,
;  122          0255  1      DBG$CVT_DIVH2_R1: JSB_R1 NOVALUE,
;  123          0256  1      DBG$CVT_DIVP_R1: JSB_R6 NOVALUE,
;  124          0257  1      DBG$CVT_MULD2_R1: JSB_R1 NOVALUE,
;  125          0258  1      DBG$CVT_MULH2_R1: JSB_R1 NOVALUE,
;  126          0259  1      DBG$CVT_MULP_R1: JSB_R6 NOVALUE,
;  127          0260  1      DBG$GET_SET_TYPEID,
;  128          0261  1      DBG$INS_ENCODE,
;  129          0262  1      DBG$MAP_DTYPE_CLASS,
;  130          0263  1      DBG$PERFORM_TYPEID_CHECK,
;  131          0264  1      DBG$STA_TYP_SUBRNG:  NOVALUE,
;  132          0265  1      DBG$STA_TYP_ATOMIC:  NOVALUE,
;  133          0266  1      DBG$STRIP_ZEROES,
;  134          0267  1      FOR$CVT_D_TE,
;  135          0268  1      FOR$CVT_D_TF,
;  136          0269  1      FOR$CVT_G_TE,
;  137          0270  1      FOR$CVT_G_TF,
;  138          0271  1      FOR$CVT_H_TE,
;  139          0272  1      FOR$CVT_H_TF,
;  140          0273  1      DBG$CVT_SCALE_OU_UP_BY_10_R1: JSB_R0 NOVALUE,
;  141          0274  1      DBG$CVT_SCALE_OU_DOWN_BY_10_R1: JSB_R0 NOVALUE,
;  142          0275  1      LIB$$CVT_SCALE_OU_UP_BY_10_R1: JSB_R0 NOVALUE,
;  143          0276  1      LIB$$CVT_SCALE_OU_DOWN_BY_10_R1: JSB_R0 NOVALUE,
;  144          0277  1      DBG$CVT_SCALE_OU_UP_BY_2_R1: JSB_R0 NOVALUE,
;  145          0278  1      DBG$CVT_SCALE_OU_DOWN_BY_2_R1: JSB_R0 NOVALUE,
;  146          0279  1      LIB$MATCH_COND,
;  147          0280  1      LIB$SIG_TO_RET: NOVALUE,
;  148          0281  1      LIB$SCOPY_R_DX6: SCOPYR_JSB_R6,
;  149          0282  1      LIB$SCOPY_DXDX6: SCOPY_JSB_R6,
;  150          0283  1      LIB$STOP: NOVALUE,
;  151          0284  1      MTH$CVT_D_G: NOVALUE,
;  152          0285  1      OTS$CVT_L_TI,
;  153          0286  1      OTS$CVT_T_D,
;  154          0287  1      OTS$CVT_T_G,
;  155          0288  1      OTS$CVT_T_H,
;  156          0289  1      SYS$ASCTIM,                                                              !
;  157          0290  1      SYS$BINTIM;                                                              !
;  158          0291  1
;  159          0292  1  EXTERNAL
;  160          0293  1      LIB$AB_CVTTP_U,                     ! These are the translation tables
;  161          0294  1      LIB$AB_CVT_O_U,                     ! used when translating to or from
;  162          0295  1      LIB$AB_CVTTP_O,                     ! packed decimal.
;  163          0296  1      LIB$AB_CVT_U_O,
;  164          0297  1      LIB$AB_CVTPT_U,
;  165          0298  1      LIB$AB_CVTPT_O,
;  166          0299  1      LIB$AB_CVTPT_Z,
;  167          0300  1      LIB$AB_CVTTP_Z,
;  168          0301  1      DBG$GL_OPCODE_NAME:  REF VECTOR[, BYTE];    ! Used in error messages.
;  169          0302  1
;  170          0303  1  EXTERNAL LITERAL
;  171          0304  1     LIB$_STRTRU;                         ! String truncated.
```

```
;  172          0305  1
;  173          0306  1 BUILTIN
;  174          0307  1         CVTTP,
;  175          0308  1         CVTSP,
;  176          0309  1         CVTLF,
;  177          0310  1         CVTLD,
;  178          0311  1         CVTPT,
;  179          0312  1         CVTPS,
;  180          0313  1         CMPP,
;  181          0314  1         CMPD,
;  182          0315  1         CVTDL,
;  183          0316  1         CVTHL,
;  184          0317  1         CVTRDL,
;  185          0318  1         CVTRFL,
;  186          0319  1         CVTDF,
;  187          0320  1         CVTPL,
;  188          0321  1         CVTLP,
;  189          0322  1         BICPSW,
;  190          0323  1         BISPSW,
;  191          0324  1         TESTBITSC,
;  192          0325  1         SUBM,
;  193          0326  1         MOVP;
;  194          0327  1
;  195          0328  1 OWN
;  196          0329  1         DECIMAL_CONVERT,        ! Tells if this is a packed decimal conversion.
;  197          0330  1                                 !   Needed so that "conversion error" can be
;  198          0331  1                                 !   signalled rather than "reserved operand".
;  199          0332  1         SAVE_RESULT;            ! Used when signalling underflow.
```

```
;  201          0333  1 ! Literals.
;  202          0334  1 !
;  203          0335  1 LITERAL
;  204          0336  1
;  205          0337  1
;  206          0338  1     ! Some general values:
;  207          0339  1     !
;  208          0340  1     K_FIRST_LONGWORD = 0,                   ! Position of first longword in buffer.
;  209          0341  1     K_INTMED_DATA_LENGTH = 32,              ! Intermediate data buffer length
;  210          0342  1     K_OUTPUT_BUFFER_LENGTH = 32,            ! Output buffer length
;  211          0343  1     K_LRGST_WU = 65535,                     ! Largest unsigned word.
;  212          0344  1     K_LRGST_LU = 4294967295,                ! Largest unsigned longword.
;  213          0345  1     K_LRGST_NEG_L = -2147483648,            ! Largest negative longword.
;  214          0346  1     K_LRGCLSSUP = DSC$K_CLASS_UBS,          ! Largest CLASS supported by routine
;  215          0347  1     K_SMLCLSSUP = DSC$K_CLASS_S,            ! Smallest CLASS supported by routine
;  216          0348  1     K_MAX_DATA_TYPES = 43,                  ! Total number of DATA TYPES in the standard
;  217          0349  1     K_MAX_CLASSES = 15,                     ! Total number of classes supported,
;  218          0350  1                                             ! including the error case 0.
;  219          0351  1     K_MIN_CLASS = DSC$K_CLASS_S,            ! Smallest class supported.
;  220          0352  1     K_MAX_CLASS = DSC$K_CLASS_UBS,          ! Largest class supported.
;  221          0353  1     K_MAX_CLASS_STA = DSC$K_CLASS_UBA,      ! Max. class number supported by standard.
;  222          0354  1     K_MIN_DTYPE_STA = DSC$K_DTYPE_V,        ! Min. data type number supported by standard.
;  223          0355  1     K_MAX_DTYPE_STA = DSC$K_DTYPE_SVU,      ! Max. data type number supported by standard.
;  224          0356  1     K_ACTUAL_CLASSES = 7,                   ! Total classes that are allowed by the STATES table.
;  225          0357  1     K_MSTNEGERR = -7,                       ! Most negative error state
;  226          0358  1     K_LRGST_NEG_B = -128,                   ! Largest negative signed byte.
;  227          0359  1     K_LRGST_NEG_W = -32768,                 ! Largest negative signed word.
;  228          0360  1     K_LRGST_B = 127,                        ! Largest positive signed byte.
;  229          0361  1     K_LRGST_W = 32767,                      ! Largest positive signed word.
;  230          0362  1     K_LRGST_BU = 255,                       ! Largest unsigned byte.
;  231          0363  1     K_LRGST_L = 2147483647,                 ! Largest signed longword.
;  232          0364  1     K_PACK_DU_LEN = 10,
;  233          0365  1
;  234          0366  1
;  235          0367  1     ! Status returned by FIND_CVT_PATH.
;  236          0368  1     !
;  237          0369  1     K_UNSCLAROU = -1,                       ! Unsupported CLASS by routine.
;  238          0370  1     K_UNSDTYROU = -2,                       ! Unsupported DATA TYPE by routine.
;  239          0371  1     K_UNSDESROU = -3,                       ! Unsupported descriptor by routine.
;  240          0372  1     K_UNSDESSTA = -4,                       ! Unsupported descriptor by standard.
;  241          0373  1     K_UNSCLASTA = -5,                       ! Unsupported CLASS by standard.
;  242          0374  1     K_UNSDTYSTA = -6,                       ! Unsupported DTYPE by standard
;  243          0375  1     K_INVNBDS = -7,                         ! Invalid NBDS
;  244          0376  1                                             ! because either array size is larger
;  245          0377  1                                             ! than a WU or it is not a one
;  246          0378  1                                             ! dimensional array.
;  247          0379  1     K_SUPPORTED = 1,                        ! This descriptor is supported, and valid.
;  248          0380  1
;  249          0381  1
;  250          0382  1     ! These are the values of the members of K_ACTUAL_CLASSES:
;  251          0383  1     ! (K_ACTUAL_CLASSES being those classes that not only exist, but are
;  252          0384  1     ! actually used by the caller.  Any other valid classes not in this
;  253          0385  1     ! group are presently treated as an error condition.)
;  254          0386  1     !
;  255          0387  1     K_STATE1_CLASS_S = DSC$K_CLASS_S,
;  256          0388  1     K_STATE2_CLASS_D = DSC$K_CLASS_D,
;  257          0389  1     K_STATE4_CLASS_A = DSC$K_CLASS_A,
```

```
;  258          0390  1     K_STATE9_CLASS_SD = DSC$K_CLASS_SD,
;  259          0391  1     K_STATE10_CLASS_NCA = DSC$K_CLASS_NCA,
;  260          0392  1     K_STATE11_CLASS_VS = DSC$K_CLASS_VS,
;  261          0393  1     K_STATE13_CLASS_UBS = DSC$K_CLASS_UBS,
;  262          0394  1
;  263          0395  1
;  264          0396  1     ! These are the intermediate data type groupings.  All data types fit into
;  265          0397  1     ! one of these groups.  The groups can represent either the left or right
;  266          0398  1     ! hand side of the conversion index.  When combined together (as in
;  267          0399  1     ! K_SMLINT_SMLINT - convert small integer to small integer, eg. byte to
;  268          0400  1     ! word), they represent the current state.
;  269          0401  1     !
;  270          0402  1     K_SMLINT = 1,
;  271          0403  1     K_LRGINT = 2,
;  272          0404  1     K_SMLFLT_CMPLX = 3,
;  273          0405  1     K_LRGFLT_CMPLX = 4,
;  274          0406  1     K_DEC = 5,
;  275          0407  1     K_NBDS = 6,
;  276          0408  1
;  277          0409  1     K_TOT_CAT = 6,
;  278          0410  1
;  279          0411  1
;  280          0412  1     ! These are the index values to the main CASE statement in DBG$CVT_DX_DX.
;  281          0413  1     !
;  282          0414  1     K_SMLINT_SMLINT = 1,
;  283          0415  1     K_SMLINT_LRGINT = 2,
;  284          0416  1     K_SMLINT_SMLFLTCMPLX = 3,
;  285          0417  1     K_SMLINT_LRGFLTCMPLX = 4,
;  286          0418  1     K_SMLINT_DEC = 5,
;  287          0419  1     K_SMLINT_NBDS = 6,
;  288          0420  1     K_LRGINT_SMLINT = 7,
;  289          0421  1     K_LRGINT_LRGINT = 8,
;  290          0422  1     K_LRGINT_SMLFLTCMPLX = 9,
;  291          0423  1     K_LRGINT_LRGFLTCMPLX = 10,
;  292          0424  1     K_LRGINT_DEC = 11,
;  293          0425  1     K_LRGINT_NBDS = 12,
;  294          0426  1     K_SMLFLTCMPLX_SMLINT = 13,
;  295          0427  1     K_SMLFLTCMPLX_LRGINT = 14,
;  296          0428  1     K_SMLFLTCMPLX_SMLFLTCMPLX = 15,
;  297          0429  1     K_SMLFLTCMPLX_LRGFLTCMPLX = 16,
;  298          0430  1     K_SMLFLTCMPLX_DEC = 17,
;  299          0431  1     K_SMLFLTCMPLX_NBDS = 18,
;  300          0432  1     K_LRGFLTCMPLX_SMLINT = 19,
;  301          0433  1     K_LRGFLTCMPLX_LRGINT = 20,
;  302          0434  1     K_LRGFLTCMPLX_SMLFLTCMPLX = 21,
;  303          0435  1     K_LRGFLTCMPLX_LRGFLTCMPLX = 22,
;  304          0436  1     K_LRGFLTCMPLX_DEC = 23,
;  305          0437  1     K_LRGFLTCMPLX_NBDS = 24,
;  306          0438  1     K_DEC_SMLINT = 25,
;  307          0439  1     K_DEC_LRGINT = 26,
;  308          0440  1     K_DEC_SMLFLTCMPLX = 27,
;  309          0441  1     K_DEC_LRGFLTCMPLX = 28,
;  310          0442  1     K_DEC_DEC = 29,
;  311          0443  1     K_DEC_NBDS = 30,
;  312          0444  1     K_NBDS_SMLINT = 31,
;  313          0445  1     K_NBDS_LRGINT = 32,
;  314          0446  1     K_NBDS_SMLFLTCMPLX = 33,
```

```
;  315          0447  1     K_NBDS_LRGFLTCMPLX = 34,
;  316          0448  1     K_NBDS_DEC = 35,
;  317          0449  1     K_NBDS_NBDS = 36,
;  318          0450  1
;  319          0451  1
;  320          0452  1     ! Length of source and destination info records in bytes.
;  321          0453  1     ! The info records are used to hold information gathered in FIND_CVT_PATH,
;  322          0454  1     ! so that it can be easily recovered and used by DBG$CVT_DX_DX.
;  323          0455  1     !
;  324          0456  1     K_SRC_INFO_LENGTH = 8,
;  325          0457  1     K_DST_INFO_LENGTH = 8,
;  326          0458  1     K_TEMP_BUF_LENGTH = 50,
;  327          0459  1
;  328          0460  1
;  329          0461  1     ! Define bit patterns for calling OTS conversion routines.
;  330          0462  1     !
;  331          0463  1     K_IGN_BLKS = 1,
;  332          0464  1     K_ENB_UNDERFLOW = 4,
;  333          0465  1     K_IGN_TABS = 16,
;  334          0466  1     K_ENB_SCALE = 64,
;  335          0467  1
;  336          0468  1
;  337          0469  1     ! Bit map to use to set all arithmetic traps
;  338          0470  1     !
;  339          0471  1     K_SET_ARITHMETIC_TRAP = 32 + 64 + 128;
```

```
;   341          0472  1 ! Macros.
;   342          0473  1 !
;   343          0474  1 MACRO
;   344          0475  1
;   345          0476  1
;   346          0477  1     ! These macros define portions of intermediate data buffer.
;   347          0478  1     !
;   348          0479  1     LONG_1 =   0, 0, 32, 0 %,
;   349          0480  1     LONG_2 =   4, 0, 32, 0 %,
;   350          0481  1     LONG_3 =   8, 0, 32, 0 %,
;   351          0482  1     LONG_4 =  12, 0, 32, 0 %,
;   352          0483  1     LONG_5 =  16, 0, 32, 0 %,
;   353          0484  1     LONG_6 =  20, 0, 32, 0 %,
;   354          0485  1     LONG_7 =  24, 0, 32, 0 %,
;   355          0486  1     LONG_8 =  28, 0, 32, 0 %,
;   356          0487  1     S_LONG_1 = 0, 0, 32, 1 %,
;   357          0488  1     S_LONG_2 = 4, 0, 32, 1 %,
;   358          0489  1     S_BYTE_1 = 0, 0,  8, 1 %,
;   359          0490  1     BYTE_1 =   0, 0,  8, 0 %,
;   360          0491  1     BYTE_2 =   1, 0,  8, 0 %,
;   361          0492  1     S_WORD_1 = 0, 0, 16, 1 %,
;   362          0493  1     WORD_1 =   0, 0, 16, 0 %,
;   363          0494  1     WORD_2 =   2, 0, 16, 0 %,
;   364          0495  1     NIBBLE_1 = 0, 0,  4, 0 %,
;   365          0496  1
;   366          0497  1
;   367          0498  1     ! This macro calculates final states given the current state and the token.
;   368          0499  1     !
;   369        M 0500  1     FINAL_STATE (CLASS, DATA_TYPE) =
;   370        M 0501  1         ((%K_MAX_DATA_TYPES *
;   371        M 0502  1         BEGIN
;   372        M 0503  1             CASE CLASS FROM K_MIN_CLASS TO K_MAX_CLASS OF
;   373        M 0504  1                 SET
;   374        M 0505  1
;   375        M 0506  1                 ! These are presenty the only classes permitted.
;   376        M 0507  1                 !
;   377        M 0508  1                 [K_STATE1_CLASS_S]:     0;
;   378        M 0509  1                 [K_STATE2_CLASS_D]:     1;
;   379        M 0510  1                 [K_STATE4_CLASS_A]:     2;
;   380        M 0511  1                 [K_STATE9_CLASS_SD]:    3;
;   381        M 0512  1                 [K_STATE10_CLASS_NCA]: 4;
;   382        M 0513  1                 [K_STATE11_CLASS_VS]:   5;
;   383        M 0514  1                 [K_STATE13_CLASS_UBS]: 6;
;   384        M 0515  1                 [INRANGE]:
;   385        M 0516  1                     BEGIN
;   386        M 0517  1                     $DBG_ERROR ('DBGCVTDX:  invalid class');
;   387        M 0518  1                     0
;   388        M 0519  1                     END;
;   389        M 0520  1                 TES
;   390          0521  1         END ) + DATA_TYPE) %,
;   391          0522  1
;   392          0523  1
;   393          0524  1     ! Again, the SRC and DST_INFO records are filled in by FIND_CVT_PATH
;   394          0525  1     ! so that information concerning the source and/or destination descriptors
;   395          0526  1     ! is readily available to DBG$CVT_DX_DX.
;   396          0527  1     !
;   397          0528  1     ! These macros are used for SRC_INFO or DST_INFO scale fields.
```

```
;  398          0529  1      !
;  399          0530  1      M_SCALE = 0, 0, 8, 1 %,
;  400          0531  1      M_BIN_SCALE = 7, 1, 1, 0 %,
;  401          0532  1
;  402          0533  1
;  403          0534  1      ! This macro is used for SRC_INFO or DST_INFO length field.
;  404          0535  1      !
;  405          0536  1      M_LEN = 5, 0, 16, 0 %,
;  406          0537  1
;  407          0538  1
;  408          0539  1      ! Define the start state.
;  409          0540  1      !
;  410          0541  1      START_STATE = VECTOR [K_MAX_CLASSES, BYTE, SIGNED] %,
;  411          0542  1
;  412          0543  1
;  413          0544  1      ! These MACROs are defined for the purpose of clarity, less typing, and anticipation
;  414          0545  1      ! of future support of BUILTINs.
;  415          0546  1      !
;  416          0547  1      ASHP    = DBG$CVT_ASHP_R1 %,
;  417          0548  1      CMPH    = DBG$CVT_CMPH_R1 %,
;  418          0549  1      CVTDH   = DBG$CVT_CVTDH_R1 %,
;  419          0550  1      CVTHD   = DBG$CVT_CVTHD_R1 %,
;  420          0551  1      CVTHF   = DBG$CVT_CVTHF_R1 %,
;  421          0552  1      CVTHG   = DBG$CVT_CVTHG_R1 %,
;  422          0553  1      CVTGH   = DBG$CVT_CVTGH_R1 %,
;  423          0554  1      CVTLB   = DBG$CVT_CVTLB_R1 %,
;  424          0555  1      CVTLH   = DBG$CVT_CVTLH_R1 %,
;  425          0556  1      CVTLW   = DBG$CVT_CVTLW_R1 %,
;  426          0557  1      CVTRDQ  = DBG$CVT_CVTRDQ_R1 %,
;  427          0558  1      CVTRHL  = DBG$CVT_CVTRHL_R1 %,
;  428          0559  1      CVTRHO  = DBG$CVT_CVTRHO_R1 %,
;  429          0560  1      CVTRHQ  = DBG$CVT_CVTRHQ_R1 %,
;  430          0561  1      CVTROUD = DBG$CVT_CVTROUD_R1 %,
;  431          0562  1      CVTROUH = DBG$CVT_CVTROUH_R1 %,
;  432          0563  1      DIVD2   = DBG$CVT_DIVD2_R1 %,
;  433          0564  1      DIVH2   = DBG$CVT_DIVH2_R1 %,
;  434          0565  1      DIVP    = DBG$CVT_DIVP_R1 %,
;  435          0566  1      MULD2   = DBG$CVT_MULD2_R1 %,
;  436          0567  1      MULH2   = DBG$CVT_MULH2_R1 %,
;  437          0568  1      MULP    = DBG$CVT_MULP_R1 %,
;  438          0569  1
;  439          0570  1
;  440          0571  1  ! The following macros scale the intermediate data.
;  441          0572  1  !
;  442          0573  1  ! These macros scale the longword in INTMED_DATA buffer.
;  443          0574  1  !
;  444        M 0575  1  M_SCALE_L_L =
;  445        M 0576  1      WHILE .BIN_SCALE GTR 0 DO
;  446        M 0577  1          BEGIN
;  447        M 0578  1          INTMED_DATA [LONG_1] = .INTMED_DATA [S_LONG_1]*2;
;  448        M 0579  1          BIN_SCALE = .BIN_SCALE - 1;
;  449        M 0580  1          END;
;  450        M 0581  1      WHILE .BIN_SCALE LSS 0 DO
;  451        M 0582  1          BEGIN
;  452        M 0583  1          INTMED_DATA [LONG_1] = .INTMED_DATA [S_LONG_1]/2;
;  453        M 0584  1          BIN_SCALE = .BIN_SCALE + 1;
;  454        M 0585  1          END;
```

```
;  455          M 0586  1      WHILE .SCALE GTR 0 DO
;  456          M 0587  1          BEGIN
;  457          M 0588  1          INTMED_DATA [LONG_1] = .INTMED_DATA [S_LONG_1]*10;
;  458          M 0589  1          SCALE = .SCALE - 1;
;  459          M 0590  1          END;
;  460          M 0591  1      WHILE .SCALE LSS 0 DO
;  461          M 0592  1          BEGIN
;  462          M 0593  1          INTMED_DATA [LONG_1] = .INTMED_DATA [S_LONG_1]/10;
;  463          M 0594  1          SCALE = .SCALE + 1;
;  464          M 0595  1          END
;  465            0596  1      %.
;  466            0597  1
;  467            0598  1
;  468            0599  1      ! Convert L to OU and scale it.  INTMED_DATA is used for L and OU
;  469            0600  1      !
;  470          M 0601  1      M_SCALE_L_OU =
;  471          M 0602  1
;  472          M 0603  1          IF .INTMED_DATA [S_LONG_1] LSS 0
;  473          M 0604  1          THEN
;  474          M 0605  1              BEGIN
;  475          M 0606  1              INTMED_DATA [LONG_1] = ABS (.INTMED_DATA [S_LONG_1]);
;  476          M 0607  1              SRC_INFO [S_SIGN] = 1;
;  477          M 0608  1              END;
;  478          M 0609  1
;  479          M 0610  1          WHILE .SCALE GTR 0 DO
;  480          M 0611  1              BEGIN
;  481          M 0612  1              LIB$$CVT_SCALE_OU_UP_BY_10_R1 (INTMED_DATA);
;  482          M 0613  1              SCALE = .SCALE - 1;
;  483          M 0614  1              END;
;  484          M 0615  1
;  485          M 0616  1          WHILE .SCALE LSS 0 DO
;  486          M 0617  1              BEGIN
;  487          M 0618  1              LIB$$CVT_SCALE_OU_DOWN_BY_10_R1 (INTMED_DATA);
;  488          M 0619  1              SCALE = .SCALE + 1;
;  489          M 0620  1              END;
;  490          M 0621  1
;  491          M 0622  1          WHILE .BIN_SCALE GTR 0 DO
;  492          M 0623  1              BEGIN
;  493          M 0624  1              DBG$CVT_SCALE_OU_UP_BY_2_R1 (INTMED_DATA);
;  494          M 0625  1              BIN_SCALE = .BIN_SCALE - 1;
;  495          M 0626  1              END;
;  496          M 0627  1
;  497          M 0628  1          WHILE .BIN_SCALE LSS 0 DO
;  498          M 0629  1              BEGIN
;  499          M 0630  1              DBG$CVT_SCALE_OU_DOWN_BY_2_R1 (INTMED_DATA);
;  500          M 0631  1              BIN_SCALE = .BIN_SCALE + 1;
;  501          M 0632  1              END
;  502          M 0633  1
;  503            0634  1      %.
;  504            0635  1
;  505            0636  1
;  506            0637  1      ! Convert L to D, and scale it.  INTMED_DATA buffer is used for L and D.
;  507            0638  1      !
;  508          M 0639  1      M_SCALE_L_D =
;  509          M 0640  1          CVTLD (INTMED_DATA, INTMED_DATA);
;  510          M 0641  1
;  511          M 0642  1          WHILE .BIN_SCALE GTR 0 DO
```

```
;   512          M 0643  1              BEGIN
;   513          M 0644  1              MULD2 (UPLIT (%D'2'), INTMED_DATA);
;   514          M 0645  1              BIN_SCALE = .BIN_SCALE - 1;
;   515          M 0646  1              END;
;   516          M 0647  1
;   517          M 0648  1          WHILE .BIN_SCALE LSS 0 DO
;   518          M 0649  1              BEGIN
;   519          M 0650  1              DIVD2 (UPLIT (%D'2'), INTMED_DATA);
;   520          M 0651  1              BIN_SCALE = .BIN_SCALE + 1;
;   521          M 0652  1              END;
;   522          M 0653  1
;   523          M 0654  1          WHILE .SCALE GTR 0 DO
;   524          M 0655  1              BEGIN
;   525          M 0656  1              MULD2 (UPLIT (%D'10'), INTMED_DATA);
;   526          M 0657  1              SCALE = .SCALE - 1;
;   527          M 0658  1              END;
;   528          M 0659  1
;   529          M 0660  1          WHILE .SCALE LSS 0 DO
;   530          M 0661  1              BEGIN
;   531          M 0662  1              DIVD2 (UPLIT (%D'10'), INTMED_DATA);
;   532          M 0663  1              SCALE = .SCALE + 1;
;   533          M 0664  1              END
;   534          M 0665  1
;   535            0666  1      %,
;   536            0667  1
;   537            0668  1
;   538            0669  1      ! Convert L to P, and scale it.  INTMED_DATA is the buffer for L and P.
;   539            0670  1      !
;   540          M 0671  1      M_SCALE_L_P =
;   541          M 0672  1
;   542          M 0673  1          IF .INTMED_DATA [S_LONG_1] LSS 0 THEN SRC_INFO [S_SIGN] = 1;
;   543          M 0674  1
;   544          M 0675  1          NO_DIGITS = 31;
;   545          M 0676  1          CVTLP (INTMED_DATA, NO_DIGITS, INTMED_DATA);
;   546          M 0677  1
;   547          M 0678  1          IF .SCALE NEQ 0
;   548          M 0679  1          THEN
;   549          M 0680  1              BEGIN
;   550          M 0681  1              MOVP (NO_DIGITS, INTMED_DATA, TEMP_BUF1);
;   551          M 0682  1              IF .CVT_ROUND_FLAG
;   552          M 0683  1              THEN
;   553          M 0684  1                  ASHP (SCALE, NO_DIGITS, TEMP_BUF1, %REF (5), NO_DIGITS, INTMED_DATA)
;   554          M 0685  1              ELSE
;   555          M 0686  1                  ASHP (SCALE, NO_DIGITS, TEMP_BUF1, %REF (0), NO_DIGITS, INTMED_DATA);
;   556          M 0687  1
;   557          M 0688  1              END;
;   558          M 0689  1
;   559          M 0690  1          WHILE .BIN_SCALE GTR 0 DO
;   560          M 0691  1              BEGIN
;   561          M 0692  1              MOVP (NO_DIGITS, INTMED_DATA, TEMP_BUF1);
;   562          M 0693  1              MULP (%REF (1), UPLIT (%P'2'), NO_DIGITS, TEMP_BUF1, NO_DIGITS, INTMED_DATA);
;   563          M 0694  1              BIN_SCALE = .BIN_SCALE - 1;
;   564          M 0695  1              END;
;   565          M 0696  1
;   566          M 0697  1          WHILE .BIN_SCALE LSS 0 DO
;   567          M 0698  1              BEGIN
;   568          M 0699  1              MOVP (NO_DIGITS, INTMED_DATA, TEMP_BUF1);
```

```
;  569          M 0700  1              DIVP (%REF (1), UPLIT (%P'2'), NO_DIGITS, TEMP_BUF1, NO_DIGITS, INTMED_DATA);
;  570          M 0701  1              BIN_SCALE = .BIN_SCALE + 1;
;  571          M 0702  1              END
;  572          M 0703  1
;  573            0704  1      %,
;  574            0705  1
;  575            0706  1
;  576            0707  1      ! Scale the OU in INTMED_DATA buffer.
;  577            0708  1      !
;  578          M 0709  1      M_SCALE_OU_OU =
;  579          M 0710  1
;  580          M 0711  1          WHILE .SCALE GTR 0 DO
;  581          M 0712  1              BEGIN
;  582          M 0713  1              LIB$$CVT_SCALE_OU_UP_BY_10_R1 (INTMED_DATA);
;  583          M 0714  1              SCALE = .SCALE - 1;
;  584          M 0715  1              END;
;  585          M 0716  1
;  586          M 0717  1          WHILE .SCALE LSS 0 DO
;  587          M 0718  1              BEGIN
;  588          M 0719  1              LIB$$CVT_SCALE_OU_DOWN_BY_10_R1 (INTMED_DATA);
;  589          M 0720  1              SCALE = .SCALE + 1;
;  590          M 0721  1              END;
;  591          M 0722  1
;  592          M 0723  1          WHILE .BIN_SCALE GTR 0 DO
;  593          M 0724  1              BEGIN
;  594          M 0725  1              DBG$CVT_SCALE_OU_UP_BY_2_R1 (INTMED_DATA);
;  595          M 0726  1              BIN_SCALE = .BIN_SCALE - 1;
;  596          M 0727  1              END;
;  597          M 0728  1
;  598          M 0729  1          WHILE .BIN_SCALE LSS 0 DO
;  599          M 0730  1              BEGIN
;  600          M 0731  1              DBG$CVT_SCALE_OU_DOWN_BY_2_R1 (INTMED_DATA);
;  601          M 0732  1              BIN_SCALE = .BIN_SCALE + 1;
;  602          M 0733  1              END
;  603          M 0734  1
;  604            0735  1      %,
;  605            0736  1
;  606            0737  1
;  607            0738  1      ! Convert OU to D, and scale it.  INTMED_DATA is used for OU and D.
;  608            0739  1      !
;  609          M 0740  1      M_SCALE_OU_D =
;  610          M 0741  1          CVTROUD (INTMED_DATA, TEMP_BUF1);
;  611          M 0742  1          CH$MOVE (8, TEMP_BUF1, INTMED_DATA);
;  612          M 0743  1
;  613          M 0744  1          WHILE .BIN_SCALE GTR 0 DO
;  614          M 0745  1              BEGIN
;  615          M 0746  1              MULD2 (UPLIT (%D'2'), INTMED_DATA);
;  616          M 0747  1              BIN_SCALE = .BIN_SCALE - 1;
;  617          M 0748  1              END;
;  618          M 0749  1
;  619          M 0750  1          WHILE .BIN_SCALE LSS 0 DO
;  620          M 0751  1              BEGIN
;  621          M 0752  1              DIVD2 (UPLIT (%D'2'), INTMED_DATA);
;  622          M 0753  1              BIN_SCALE = .BIN_SCALE + 1;
;  623          M 0754  1              END;
;  624          M 0755  1
;  625          M 0756  1          WHILE .SCALE GTR 0 DO
```

```
;  626          M 0757  1              BEGIN
;  627          M 0758  1              MULD2 (UPLIT (%D'10'), INTMED_DATA);
;  628          M 0759  1              SCALE = .SCALE - 1;
;  629          M 0760  1              END;
;  630          M 0761  1
;  631          M 0762  1          WHILE .SCALE LSS 0 DO
;  632          M 0763  1              BEGIN
;  633          M 0764  1              DIVD2 (UPLIT (%D'10'), INTMED_DATA);
;  634          M 0765  1              SCALE = .SCALE + 1;
;  635          M 0766  1              END
;  636          M 0767  1
;  637            0768  1      %,
;  638            0769  1
;  639            0770  1
;  640            0771  1      ! Convert OU to H, and scale it.  INTMED_DATA is used for OU and H.
;  641            0772  1      !
;  642          M 0773  1      M_SCALE_OU_H =
;  643          M 0774  1          CVTROUH (INTMED_DATA, TEMP_BUF1);
;  644          M 0775  1          CH$MOVE (16, TEMP_BUF1, INTMED_DATA);
;  645          M 0776  1
;  646          M 0777  1          WHILE .BIN_SCALE GTR 0 DO
;  647          M 0778  1              BEGIN
;  648          M 0779  1              MULH2 (UPLIT (%H'2'), INTMED_DATA);
;  649          M 0780  1              BIN_SCALE = .BIN_SCALE - 1;
;  650          M 0781  1              END;
;  651          M 0782  1
;  652          M 0783  1          WHILE .BIN_SCALE LSS 0 DO
;  653          M 0784  1              BEGIN
;  654          M 0785  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA);
;  655          M 0786  1              BIN_SCALE = .BIN_SCALE + 1;
;  656          M 0787  1              END;
;  657          M 0788  1
;  658          M 0789  1          WHILE .SCALE GTR 0 DO
;  659          M 0790  1              BEGIN
;  660          M 0791  1              MULH2 (UPLIT (%H'10'), INTMED_DATA);
;  661          M 0792  1              SCALE = .SCALE - 1;
;  662          M 0793  1              END;
;  663          M 0794  1
;  664          M 0795  1          WHILE .SCALE LSS 0 DO
;  665          M 0796  1              BEGIN
;  666          M 0797  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA);
;  667          M 0798  1              SCALE = .SCALE + 1;
;  668          M 0799  1              END
;  669          M 0800  1
;  670            0801  1      %,
;  671            0802  1
;  672            0803  1
;  673            0804  1      ! Convert L to H, and scale it.  INTMED_DATA is used for L and H.
;  674            0805  1      !
;  675          M 0806  1      M_SCALE_L_H =
;  676          M 0807  1          CVTLH (INTMED_DATA, INTMED_DATA);
;  677          M 0808  1
;  678          M 0809  1          WHILE .BIN_SCALE GTR 0 DO
;  679          M 0810  1              BEGIN
;  680          M 0811  1              MULH2 (UPLIT (%H'2'), INTMED_DATA);
;  681          M 0812  1              BIN_SCALE = .BIN_SCALE - 1;
;  682          M 0813  1              END;
```

```
;  683          M 0814  1
;  684          M 0815  1          WHILE .BIN_SCALE LSS 0 DO
;  685          M 0816  1              BEGIN
;  686          M 0817  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA);
;  687          M 0818  1              BIN_SCALE = .BIN_SCALE + 1;
;  688          M 0819  1              END;
;  689          M 0820  1
;  690          M 0821  1          WHILE .SCALE GTR 0 DO
;  691          M 0822  1              BEGIN
;  692          M 0823  1              MULH2 (UPLIT (%H'10'), INTMED_DATA);
;  693          M 0824  1              SCALE = .SCALE - 1;
;  694          M 0825  1              END;
;  695          M 0826  1
;  696          M 0827  1          WHILE .SCALE LSS 0 DO
;  697          M 0828  1              BEGIN
;  698          M 0829  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA);
;  699          M 0830  1              SCALE = .SCALE + 1;
;  700          M 0831  1              END
;  701          M 0832  1
;  702            0833  1      %,
;  703            0834  1
;  704            0835  1
;  705            0836  1      ! Scale D in INTMED_DATA.
;  706            0837  1      !
;  707          M 0838  1      M_SCALE_D_D =
;  708          M 0839  1
;  709          M 0840  1          WHILE .BIN_SCALE GTR 0 DO
;  710          M 0841  1              BEGIN
;  711          M 0842  1              MULD2 (UPLIT (%D'2'), INTMED_DATA);
;  712          M 0843  1              MULD2 (UPLIT (%D'2'), INTMED_DATA+8);
;  713          M 0844  1              BIN_SCALE = .BIN_SCALE - 1;
;  714          M 0845  1              END;
;  715          M 0846  1
;  716          M 0847  1          WHILE .BIN_SCALE LSS 0 DO
;  717          M 0848  1              BEGIN
;  718          M 0849  1              DIVD2 (UPLIT (%D'2'), INTMED_DATA);
;  719          M 0850  1              DIVD2 (UPLIT (%D'2'), INTMED_DATA+8);
;  720          M 0851  1              BIN_SCALE = .BIN_SCALE + 1;
;  721          M 0852  1              END;
;  722          M 0853  1
;  723          M 0854  1          WHILE .SCALE GTR 0 DO
;  724          M 0855  1              BEGIN
;  725          M 0856  1              MULD2 (UPLIT (%D'10'), INTMED_DATA);
;  726          M 0857  1              MULD2 (UPLIT (%D'10'), INTMED_DATA+8);
;  727          M 0858  1              SCALE = .SCALE - 1;
;  728          M 0859  1              END;
;  729          M 0860  1
;  730          M 0861  1          WHILE .SCALE LSS 0 DO
;  731          M 0862  1              BEGIN
;  732          M 0863  1              DIVD2 (UPLIT (%D'10'), INTMED_DATA);
;  733          M 0864  1              DIVD2 (UPLIT (%D'10'), INTMED_DATA+8);
;  734          M 0865  1              SCALE = .SCALE + 1;
;  735          M 0866  1              END
;  736          M 0867  1
;  737            0868  1      %,
;  738            0869  1
;  739            0870  1
```

```
;  740          0871  1      ! Convert D to H, and scale it.  INTMED_DATA is used for D and H.
;  741          0872  1      !
;  742        M 0873  1      M_SCALE_D_H =
;  743        M 0874  1          CVTDH (INTMED_DATA, TEMP_BUF1);
;  744        M 0875  1          CVTDH (INTMED_DATA+8, INTMED_DATA+16);
;  745        M 0876  1          CH$MOVE (16, TEMP_BUF1, INTMED_DATA);
;  746        M 0877  1
;  747        M 0878  1          WHILE .BIN_SCALE GTR 0 DO
;  748        M 0879  1              BEGIN
;  749        M 0880  1              MULH2 (UPLIT (%H'2'), INTMED_DATA);
;  750        M 0881  1              MULH2 (UPLIT (%H'2'), INTMED_DATA+16);
;  751        M 0882  1              BIN_SCALE = .BIN_SCALE - 1;
;  752        M 0883  1              END;
;  753        M 0884  1
;  754        M 0885  1          WHILE .BIN_SCALE LSS 0 DO
;  755        M 0886  1              BEGIN
;  756        M 0887  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA);
;  757        M 0888  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA+16);
;  758        M 0889  1              BIN_SCALE = .BIN_SCALE + 1;
;  759        M 0890  1              END;
;  760        M 0891  1
;  761        M 0892  1          WHILE .SCALE GTR 0 DO
;  762        M 0893  1              BEGIN
;  763        M 0894  1              MULH2 (UPLIT (%H'10'), INTMED_DATA);
;  764        M 0895  1              MULH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  765        M 0896  1              SCALE = .SCALE - 1;
;  766        M 0897  1              END;
;  767        M 0898  1
;  768        M 0899  1          WHILE .SCALE LSS 0 DO
;  769        M 0900  1              BEGIN
;  770        M 0901  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA);
;  771        M 0902  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  772        M 0903  1              SCALE = .SCALE + 1;
;  773        M 0904  1              END
;  774        M 0905  1
;  775          0906  1      %,
;  776          0907  1
;  777          0908  1
;  778          0909  1      ! Convert G to H, and scale it.  INTMED_DATA is used for G and H.
;  779          0910  1      !
;  780        M 0911  1      M_SCALE_G_H =
;  781        M 0912  1          BEGIN
;  782        M 0913  1          CVTGH (INTMED_DATA, TEMP_BUF1);
;  783        M 0914  1          CVTGH (INTMED_DATA+8, INTMED_DATA+16);
;  784        M 0915  1          CH$MOVE (16, TEMP_BUF1, INTMED_DATA);
;  785        M 0916  1
;  786        M 0917  1          WHILE .BIN_SCALE GTR 0 DO
;  787        M 0918  1              BEGIN
;  788        M 0919  1              MULH2 (UPLIT (%H'2'), INTMED_DATA);
;  789        M 0920  1              MULH2 (UPLIT (%H'2'), INTMED_DATA+16);
;  790        M 0921  1              BIN_SCALE = .BIN_SCALE - 1;
;  791        M 0922  1              END;
;  792        M 0923  1
;  793        M 0924  1          WHILE .BIN_SCALE LSS 0 DO
;  794        M 0925  1              BEGIN
;  795        M 0926  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA);
;  796        M 0927  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA+16);
```

```
;  797          M 0928  1              BIN_SCALE = .BIN_SCALE + 1;
;  798          M 0929  1              END;
;  799          M 0930  1
;  800          M 0931  1          WHILE .SCALE GTR 0 DO
;  801          M 0932  1              BEGIN
;  802          M 0933  1              MULH2 (UPLIT (%H'10'), INTMED_DATA);
;  803          M 0934  1              MULH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  804          M 0935  1              SCALE = .SCALE - 1;
;  805          M 0936  1              END;
;  806          M 0937  1
;  807          M 0938  1          WHILE .SCALE LSS 0 DO
;  808          M 0939  1              BEGIN
;  809          M 0940  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA);
;  810          M 0941  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  811          M 0942  1              SCALE = .SCALE + 1;
;  812          M 0943  1              END;
;  813          M 0944  1          END
;  814          M 0945  1
;  815            0946  1      %,
;  816            0947  1
;  817            0948  1
;  818            0949  1      ! Scale H in INTMED_DATA.
;  819            0950  1      !
;  820          M 0951  1      M_SCALE_H_H =
;  821          M 0952  1
;  822          M 0953  1          WHILE .BIN_SCALE GTR 0 DO
;  823          M 0954  1              BEGIN
;  824          M 0955  1              MULH2 (UPLIT (%H'2'), INTMED_DATA);
;  825          M 0956  1              MULH2 (UPLIT (%H'2'), INTMED_DATA+16);
;  826          M 0957  1              BIN_SCALE = .BIN_SCALE - 1;
;  827          M 0958  1              END;
;  828          M 0959  1
;  829          M 0960  1          WHILE .BIN_SCALE LSS 0 DO
;  830          M 0961  1              BEGIN
;  831          M 0962  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA);
;  832          M 0963  1              DIVH2 (UPLIT (%H'2'), INTMED_DATA+16);
;  833          M 0964  1              BIN_SCALE = .BIN_SCALE + 1;
;  834          M 0965  1              END;
;  835          M 0966  1
;  836          M 0967  1          WHILE .SCALE GTR 0 DO
;  837          M 0968  1              BEGIN
;  838          M 0969  1              MULH2 (UPLIT (%H'10'), INTMED_DATA);
;  839          M 0970  1              MULH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  840          M 0971  1              SCALE = .SCALE - 1;
;  841          M 0972  1              END;
;  842          M 0973  1
;  843          M 0974  1          WHILE .SCALE LSS 0 DO
;  844          M 0975  1              BEGIN
;  845          M 0976  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA);
;  846          M 0977  1              DIVH2 (UPLIT (%H'10'), INTMED_DATA+16);
;  847          M 0978  1              SCALE = .SCALE + 1;
;  848          M 0979  1              END
;  849          M 0980  1
;  850            0981  1      %,
;  851            0982  1
;  852            0983  1
;  853            0984  1      ! Scale P in INTMED_DATA
```

```
;  854          0985  1     !
;  855        M 0986  1     M_SCALE_P_P =
;  856        M 0987  1         NO_DIGITS = .SRC_INFO [S_LEN];
;  857        M 0988  1
;  858        M 0989  1         IF (CMPP (NO_DIGITS, INTMED_DATA, %REF (1), .PACK_ZERO) LSS 0) THEN SRC_INFO [S_SIGN] = 1;
;  859        M 0990  1
;  860        M 0991  1         IF .SCALE NEQ 0
;  861        M 0992  1         THEN
;  862        M 0993  1             BEGIN
;  863        M 0994  1             MOVP (NO_DIGITS, INTMED_DATA, TEMP_BUF1);
;  864        M 0995  1             IF .CVT_ROUND_FLAG
;  865        M 0996  1             THEN
;  866        M 0997  1                 ASHP (SCALE, NO_DIGITS, TEMP_BUF1, %REF (5), NO_DIGITS, INTMED_DATA)
;  867        M 0998  1             ELSE
;  868        M 0999  1                 ASHP (SCALE, NO_DIGITS, TEMP_BUF1, %REF (0), NO_DIGITS, INTMED_DATA);
;  869        M 1000  1             END;
;  870        M 1001  1
;  871        M 1002  1
;  872        M 1003  1 !           I (PS) added the following code, because, if I deposit a packed decimal
;  873        M 1004  1 !           number 999.888 into a 4 digits decimal number scaled -2, I want to get
;  874        M 1005  1 !           a result of 99.88, instead of later on I will get overflow error, and
;  875        M 1006  1 !           have nothing as result.  Check to see if the siganificant digits of
;  876        M 1007  1 !           source is greater than the siganificant digits of the destination.
;  877        M 1008  1 !
;  878        M 1009  1 !           This piece of code is used only if both operands are packed.
;  879        M 1010  1 !
;  880        M 1011  1
;  881        M 1012  1         IF (.SOURCE[DSC$W_LENGTH] + .SOURCE[DSC$B_SCALE] GTR
;  882        M 1013  1             .DESTINATION[DSC$W_LENGTH] + .DESTINATION[DSC$B_SCALE]) AND
;  883        M 1014  1            (.SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_P AND
;  884        M 1015  1             .DESTINATION[DSC$B_DTYPE] EQL DSC$K_DTYPE_P)
;  885        M 1016  1         THEN
;  886        M 1017  1             BEGIN
;  887        M 1018  1             LOCAL
;  888        M 1019  1                 HIGH_NIBBLE_PTR: REF VECTOR[,BYTE],
;  889        M 1020  1                 LOW_NIBBLE_PTR: REF VECTOR[,BYTE];
;  890        M 1021  1
;  891        M 1022  1
;  892        M 1023  1             ! Point to the last digits.
;  893        M 1024  1             !
;  894        M 1025  1             HIGH_NIBBLE_PTR = INTMED_DATA + 16 - 1;
;  895        M 1026  1
;  896        M 1027  1
;  897        M 1028  1             ! Backup the pointer to the siganificant digit
;  898        M 1029  1             ! needs to be truncated.  Zero out everything
;  899        M 1030  1             ! before that.
;  900        M 1031  1             !
;  901        M 1032  1             LOW_NIBBLE_PTR = .HIGH_NIBBLE_PTR -
;  902        M 1033  1                     (.DESTINATION[DSC$W_LENGTH] / 2 + 1) + 1;
;  903        M 1034  1
;  904        M 1035  1
;  905        M 1036  1             ! If destination digits is even, we need to
;  906        M 1037  1             ! zero out one nibble.  Note: this may be already zero.
;  907        M 1038  1             !
;  908        M 1039  1             IF (.DESTINATION[DSC$W_LENGTH] MOD 2) EQL 0
;  909        M 1040  1             THEN
;  910        M 1041  1                 LOW_NIBBLE_PTR[0] = .LOW_NIBBLE_PTR[0] AND %X'0F';
```

```
;  911          M 1042  1
;  912          M 1043  1
;  913          M 1044  1              ! Zero out everything before it.  Note: this may be already
;  914          M 1045  1              ! zero.
;  915          M 1046  1              !
;  916          M 1047  1              LOW_NIBBLE_PTR = .LOW_NIBBLE_PTR - 1;
;  917          M 1048  1              WHILE .LOW_NIBBLE_PTR GEQ INTMED_DATA DO
;  918          M 1049  1                  BEGIN
;  919          M 1050  1                  LOW_NIBBLE_PTR[0] = %X'00';
;  920          M 1051  1                  LOW_NIBBLE_PTR = .LOW_NIBBLE_PTR - 1;
;  921          M 1052  1                  END;
;  922          M 1053  1
;  923          M 1054  1              SIGNAL(DBG$_INUMTRUNC, 1, .DBG$GL_OPCODE_NAME);
;  924          M 1055  1              END
;  925          M 1056  1
;  926            1057  1      %,
;  927            1058  1
;  928            1059  1
;  929            1060  1      ! Convert P to OU, and scale it.  INTMED_DATA is used for P and OU.
;  930            1061  1      !
;  931          M 1062  1      M_SCALE_P_OU =
;  932          M 1063  1          NO_DIGITS = .SRC_INFO [S_LEN];
;  933          M 1064  1          CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF1);
;  934          M 1065  1          CLASS_S_DESC [DSC$W_LENGTH] = .NO_DIGITS + 1;
;  935          M 1066  1          CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF1;
;  936          M 1067  1          OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF2);
;  937          M 1068  1
;  938          M 1069  1          IF .TEMP_BUF2 [0, 15, 1, 0]
;  939          M 1070  1          THEN
;  940          M 1071  1              BEGIN
;  941          M 1072  1              TEMP_BUF2 [0, 15, 1, 0] = 0;
;  942          M 1073  1              SRC_INFO [S_SIGN] = 1;
;  943          M 1074  1              END;
;  944          M 1075  1
;  945          M 1076  1          CVTRHO (TEMP_BUF2, INTMED_DATA);
;  946          M 1077  1
;  947          M 1078  1          WHILE .SCALE GTR 0 DO
;  948          M 1079  1              BEGIN
;  949          M 1080  1              LIB$$CVT_SCALE_OU_UP_BY_10_R1 (INTMED_DATA);
;  950          M 1081  1              SCALE = .SCALE - 1;
;  951          M 1082  1              END;
;  952          M 1083  1
;  953          M 1084  1          WHILE .SCALE LSS 0 DO
;  954          M 1085  1              BEGIN
;  955          M 1086  1              LIB$$CVT_SCALE_OU_DOWN_BY_10_R1 (INTMED_DATA);
;  956          M 1087  1              SCALE = .SCALE + 1;
;  957          M 1088  1              END;
;  958          M 1089  1
;  959          M 1090  1          WHILE .BIN_SCALE GTR 0 DO
;  960          M 1091  1              BEGIN
;  961          M 1092  1              DBG$CVT_SCALE_OU_UP_BY_2_R1 (INTMED_DATA);
;  962          M 1093  1              BIN_SCALE = .BIN_SCALE - 1;
;  963          M 1094  1              END;
;  964          M 1095  1
;  965          M 1096  1          WHILE .BIN_SCALE LSS 0 DO
;  966          M 1097  1              BEGIN
;  967          M 1098  1              DBG$CVT_SCALE_OU_DOWN_BY_2_R1 (INTMED_DATA);
```

```
;  968          M 1099  1              BIN_SCALE = .BIN_SCALE + 1;
;  969          M 1100  1              END
;  970          M 1101  1
;  971            1102  1      %,
;  972            1103  1
;  973            1104  1
;  974            1105  1      ! Convert P to D, and scale it.  INTMED_DATA is used for P and D.
;  975            1106  1      !
;  976          M 1107  1      M_SCALE_P_D =
;  977          M 1108  1          NO_DIGITS = .SRC_INFO [S_LEN];
;  978          M 1109  1
;  979          M 1110  1          ! In the case of scaled packed, we need to get the scale this way.
;  980          M 1111  1          !
;  981          M 1112  1          IF .SOURCE[DSC$B_CLASS] EQL DSC$K_CLASS_SD
;  982          M 1113  1          THEN
;  983          M 1114  1              SCALE = - .SOURCE[DSC$B_SCALE];
;  984          M 1115  1
;  985          M 1116  1          CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF1);
;  986          M 1117  1          CLASS_S_DESC [DSC$W_LENGTH] = .NO_DIGITS + 1;
;  987          M 1118  1          CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF1;
;  988          M 1119  1          STATUS = OTS$CVT_T_D (CLASS_S_DESC, INTMED_DATA, 0, .SCALE, (K_ENB_UNDERFLOW OR K_ENB_SCALE));
;  989          M 1120  1
;  990          M 1121  1          IF NOT .STATUS
;  991          M 1122  1          THEN
;  992          M 1123  1              IF .SCALE LSS 0
;  993          M 1124  1                  THEN SIGNAL (DBG$_IFLTUND, 1, .DBG$GL_OPCODE_NAME)
;  994            1125  1                  ELSE SIGNAL (DBG$_FLTOVF, 1, .DBG$GL_OPCODE_NAME); %,
;  995            1126  1
;  996            1127  1
;  997            1128  1      ! Convert P to H, and scale it.  INTMED_DATA is used for P and H.
;  998            1129  1      !
;  999          M 1130  1      M_SCALE_P_H =
; 1000          M 1131  1          NO_DIGITS = .SRC_INFO [S_LEN];
; 1001          M 1132  1
; 1002          M 1133  1          ! In the case of scaled packed, we need to get the scale this way.
; 1003          M 1134  1          !
; 1004          M 1135  1          IF .SOURCE[DSC$B_CLASS] EQL DSC$K_CLASS_SD
; 1005          M 1136  1          THEN
; 1006          M 1137  1              SCALE = - .SOURCE[DSC$B_SCALE];
; 1007          M 1138  1
; 1008          M 1139  1          CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF1);
; 1009          M 1140  1          CLASS_S_DESC [DSC$W_LENGTH] = .NO_DIGITS + 1;
; 1010          M 1141  1          CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF1;
; 1011          M 1142  1          STATUS = OTS$CVT_T_H (CLASS_S_DESC, INTMED_DATA, 0, .SCALE, (K_ENB_UNDERFLOW OR K_ENB_SCALE));
; 1012          M 1143  1
; 1013          M 1144  1          IF NOT .STATUS
; 1014          M 1145  1          THEN
; 1015          M 1146  1              IF .SCALE LSS 0
; 1016          M 1147  1                  THEN SIGNAL (DBG$_IFLTUND, 1, .DBG$GL_OPCODE_NAME)
; 1017            1148  1                  ELSE SIGNAL (DBG$_FLTOVF, 1, .DBG$GL_OPCODE_NAME); %;
```

```
;  1019          1149  1 ! Structure and Field Definitions.
;  1020          1150  1 !
;  1021          1151  1 ! STATES is a structure into which go all the states other than the first.
;  1022          1152  1 ! The final states and the states that never get used (such as the states that
;  1023          1153  1 ! contain non-supported CLASSes) will not be in this structure.
;  1024          1154  1 !
;  1025          1155  1 STRUCTURE
;  1026          1156  1     STATES [STATE, TOKEN] =
;  1027          1157  1         [K_ACTUAL_CLASSES*K_MAX_DATA_TYPES]
;  1028          1158  3         (STATES + (K_MAX_DATA_TYPES*
;  1029          1159  4         BEGIN
;  1030          1160  4         CASE STATE FROM K_MIN_CLASS TO K_MAX_CLASS OF
;  1031          1161  4             SET
;  1032          1162  4             [K_STATE1_CLASS_S]:    0;
;  1033          1163  4             [K_STATE2_CLASS_D]:    1;
;  1034          1164  4             [K_STATE4_CLASS_A]:    2;
;  1035          1165  4             [K_STATE9_CLASS_SD]:   3;
;  1036          1166  4             [K_STATE10_CLASS_NCA]: 4;
;  1037          1167  4             [K_STATE11_CLASS_VS]:  5;
;  1038          1168  4             [K_STATE13_CLASS_UBS]: 6;
;  1039          1169  4             [INRANGE, OUTRANGE]:
;  1040          1170  5                 BEGIN
;  1041          1171  5                 $DBG_ERROR ('DBGCVTDX:  invalid class');
;  1042          1172  5                 0
;  1043          1173  4                 END;
;  1044          1174  4             TES
;  1045          1175  4         END
;  1046          1176  1         ) + TOKEN)<0, %BPUNIT, 1>;
;  1047          1177  1
;  1048          1178  1
;  1049          1179  1 ! SRC and DST INFO record fields.
;  1050          1180  1 !
;  1051          1181  1 FIELD
;  1052          1182  1     SRC_INFO_FIELDS =
;  1053          1183  1         SET
;  1054          1184  1         S_SCALE = [0, 0, 8, 1],
;  1055          1185  1         S_POINTER = [1, 0, 32, 0],
;  1056          1186  1         S_LEN = [5, 0, 16, 0],
;  1057          1187  1         S_SIGN = [7, 0, 1, 0],
;  1058          1188  1         S_BIN_SCALE = [7, 1, 1, 0]      ! Flag indicating scale is binary
;  1059          1189  1         TES;
;  1060          1190  1
;  1061          1191  1 FIELD
;  1062          1192  1     DST_INFO_FIELDS =
;  1063          1193  1         SET
;  1064          1194  1         D_SCALE = [0, 0, 8, 1],
;  1065          1195  1         D_LEN = [5, 0, 16, 0],
;  1066          1196  1         D_BIN_SCALE = [7, 1, 1, 0]      ! Flag indicating scale is binary
;  1067          1197  1         TES;
```

```
;  1069          1198  1 ! State Table.
;  1070          1199  1 !
;  1071          1200  1 ! Start States (all classes).  CLASS_TABLE.
;  1072          1201  1 ! These are the start state entries.
;  1073          1202  1 ! For each CLASS in the standard there is an entry here.  They are:
;  1074          1203  1 !       Z       ,S      ,D      ,V      ,A
;  1075          1204  1 !      ,P       ,none   ,J      ,none   ,SD
;  1076          1205  1 !      ,NCA     ,VS     ,VSA    ,UBS    ,UBA.
;  1077          1206  1 !
;  1078          1207  1 BIND
;  1079          1208  1     CLASS_TABLE = UPLIT BYTE
;  1080          1209  1     %( Start state.  All classes. )%
;  1081          1210  1     (K_UNSCLAROU,DSC$K_CLASS_S,DSC$K_CLASS_D,K_UNSCLAROU,DSC$K_CLASS_A
;  1082          1211  1     ,K_UNSCLAROU,K_UNSCLASTA,K_UNSCLAROU,K_UNSCLASTA,DSC$K_CLASS_SD
;  1083          1212  1     ,DSC$K_CLASS_NCA,DSC$K_CLASS_VS,K_UNSCLAROU,DSC$K_CLASS_UBS,K_UNSCLAROU): START_STATE;
;  1084          1213  1
;  1085          1214  1
;  1086          1215  1 ! Remaining States.  DTYPE_TABLE.
;  1087          1216  1 !
;  1088          1217  1 ! This is the rest of the state table.  It is separate for space efficiency.
;  1089          1218  1 ! Each state contains entries for each data type supported by the standard.
;  1090          1219  1 ! Note that for space efficiency the final states are not in the vector.
;  1091          1220  1 ! Also since each state represents a supported CLASS, if a CLASS is not
;  1092          1221  1 ! supported (by the standard or routine), then the state has no entry in
;  1093          1222  1 ! the vector.  The index table for the vector will index to the proper place
;  1094          1223  1 ! in the vector below.
;  1095          1224  1 ! The table below shows graphically what descriptors are valid.
;  1096          1225  1 !
;  1097          1226  1 !                                                DSC$K_DTYPE_x
;  1098          1227  1 !                      BU WU LU B W L Q F D G H T ND NL NLO NR NRO NZ P VT AC AZ TF V SV VU SVU
;  1099          1228  1 !DSC$K_CLASS_S        x  x  x  x x x x x x x x x x  x  x   x  x x  x  x                 x  x
;  1100          1229  1 !DSC$K_CLASS_D                                  x
;  1101          1230  1 !DSC$K_CLASS_SD             x x x x x x x x x x  x  x   x  x x  x  x
;  1102          1231  1 !DSC$K_CLASS_VS                                 x                          x  x  x
;  1103          1232  1 !DSC$K_CLASS_A     x                           x
;  1104          1233  1 !DSC$K_CLASS_NCA   x                           x
;  1105          1234  1 !DSC$K_CLASS_UBS                                                                 x        x   x
;  1106          1235  1 !
;  1107          1236  1 !
;  1108          1237  1 ! Note that these data types are hard coded in (zero based vector, and position
;  1109          1238  1 ! of each data type is determined be the value of the symbol), so if data type
;  1110          1239  1 ! values are ever rearranged this table must be rearranged.
;  1111          1240  1 !
;  1112          1241  1 BIND
;  1113          1242  1     DTYPE_TABLE = UPLIT BYTE
;  1114          1243  1     %( State zero.  Class z. )%
;  1115          1244  1     %( State one.  Class s. )%
;  1116          1245  1         (K_UNSDTYROU,DSC$K_DTYPE_V,DSC$K_DTYPE_BU,DSC$K_DTYPE_WU,DSC$K_DTYPE_LU
;  1117          1246  1         ,DSC$K_DTYPE_QU,DSC$K_DTYPE_B,DSC$K_DTYPE_W,DSC$K_DTYPE_L,DSC$K_DTYPE_Q
;  1118          1247  1         ,DSC$K_DTYPE_F,DSC$K_DTYPE_D,DSC$K_DTYPE_FC,DSC$K_DTYPE_DC,DSC$K_DTYPE_T
;  1119          1248  1         ,DSC$K_DTYPE_NU,DSC$K_DTYPE_NL,DSC$K_DTYPE_NLO,DSC$K_DTYPE_NR,DSC$K_DTYPE_NRO
;  1120          1249  1         ,DSC$K_DTYPE_NZ,DSC$K_DTYPE_P,DSC$K_DTYPE_ZI,K_UNSDTYROU,K_UNSDESSTA
;  1121          1250  1         ,K_UNSDTYROU,DSC$K_DTYPE_O,DSC$K_DTYPE_G,DSC$K_DTYPE_H,DSC$K_DTYPE_GC
;  1122          1251  1         ,DSC$K_DTYPE_HC,K_UNSDTYROU,K_UNSDTYROU,K_UNSDTYROU,K_UNSDTYROU
;  1123          1252  1         ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,DSC$K_DTYPE_TF
;  1124          1253  1         ,DSC$K_DTYPE_SV,K_UNSDTYROU
;  1125          1254  1     %( State two.  Class d. )%
```

```
;  1126          1255  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1127          1256  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1128          1257  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,DSC$K_DTYPE_T
;  1129          1258  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1130          1259  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1131          1260  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1132          1261  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1133          1262  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1134          1263  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1135          1264  1      %( State three.  Class v. )%
;  1136          1265  1      %( State four.  Class a. )%
;  1137          1266  1          ,K_UNSDTYROU,K_UNSDTYROU,DSC$K_DTYPE_BU,K_UNSDESROU,K_UNSDESROU
;  1138          1267  1          ,K_UNSDTYROU,K_UNSDESROU,K_UNSDESROU,K_UNSDESROU,K_UNSDESROU
;  1139          1268  1          ,K_UNSDESROU,K_UNSDESROU,K_UNSDTYROU,K_UNSDTYROU,DSC$K_DTYPE_T
;  1140          1269  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1141          1270  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDTYROU
;  1142          1271  1          ,K_UNSDTYROU,K_UNSDTYROU,K_UNSDESROU,K_UNSDESROU,K_UNSDTYROU
;  1143          1272  1          ,K_UNSDESSTA,K_UNSDTYROU,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1144          1273  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1145          1274  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1146          1275  1      %( State five.  Class p. )%
;  1147          1276  1      %( State six.  Class 'undefined' )%
;  1148          1277  1      %( State seven.  Class j. )%
;  1149          1278  1      %( State eight.  Class 'undefined' )%
;  1150          1279  1      %( State nine.  Class sd. )%
;  1151          1280  1          ,K_UNSDESSTA,K_UNSDESSTA,DSC$K_DTYPE_BU,DSC$K_DTYPE_WU,DSC$K_DTYPE_LU
;  1152          1281  1          ,DSC$K_DTYPE_QU,DSC$K_DTYPE_B,DSC$K_DTYPE_W,DSC$K_DTYPE_L,DSC$K_DTYPE_Q
;  1153          1282  1          ,DSC$K_DTYPE_F,DSC$K_DTYPE_D,DSC$K_DTYPE_FC,DSC$K_DTYPE_DC,DSC$K_DTYPE_T
;  1154          1283  1          ,DSC$K_DTYPE_NU,DSC$K_DTYPE_NL,DSC$K_DTYPE_NLO,DSC$K_DTYPE_NR,DSC$K_DTYPE_NRO
;  1155          1284  1          ,DSC$K_DTYPE_NZ,DSC$K_DTYPE_P,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1156          1285  1          ,K_UNSDESSTA,DSC$K_DTYPE_O,DSC$K_DTYPE_G,DSC$K_DTYPE_H,DSC$K_DTYPE_GC
;  1157          1286  1          ,DSC$K_DTYPE_HC,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1158          1287  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1159          1288  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1160          1289  1      %( State ten.  Class nca. )%
;  1161          1290  1          ,K_UNSDTYROU,K_UNSDTYROU,DSC$K_DTYPE_BU,K_UNSDESROU,K_UNSDESROU
;  1162          1291  1          ,K_UNSDTYROU,K_UNSDESROU,K_UNSDESROU,K_UNSDESROU,K_UNSDESROU
;  1163          1292  1          ,K_UNSDESROU,K_UNSDESROU,K_UNSDTYROU,K_UNSDTYROU,DSC$K_DTYPE_T
;  1164          1293  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1165          1294  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDTYROU
;  1166          1295  1          ,K_UNSDTYROU,K_UNSDTYROU,K_UNSDESROU,K_UNSDESROU,K_UNSDTYROU
;  1167          1296  1          ,K_UNSDESSTA,K_UNSDTYROU,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1168          1297  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1169          1298  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1170          1299  1      %( State eleven.  Class vs. )%
;  1171          1300  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1172          1301  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1173          1302  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,DSC$K_DTYPE_T
;  1174          1303  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1175          1304  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1176          1305  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1177          1306  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1178          1307  1          ,K_UNSDESSTA,K_UNSDESSTA,DSC$K_DTYPE_VT,DSC$K_DTYPE_AC
;  1179          1308  1          ,DSC$K_DTYPE_AZ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1180          1309  1      %( State twelve.  Class vsa. )%
;  1181          1310  1      %( State thirteen.  Class ubs. )%
;  1182          1311  1          ,K_UNSDTYROU,DSC$K_DTYPE_V,DSC$K_DTYPE_BU,DSC$K_DTYPE_WU,DSC$K_DTYPE_LU
```

```
;  1183          1312  1          ,DSC$K_DTYPE_QU,DSC$K_DTYPE_B,DSC$K_DTYPE_W,DSC$K_DTYPE_L,DSC$K_DTYPE_Q
;  1184          1313  1          ,DSC$K_DTYPE_F,DSC$K_DTYPE_D,DSC$K_DTYPE_FC,DSC$K_DTYPE_DC,DSC$K_DTYPE_T
;  1185          1314  1          ,DSC$K_DTYPE_NU,DSC$K_DTYPE_NL,DSC$K_DTYPE_NLO,DSC$K_DTYPE_NR,DSC$K_DTYPE_NRO
;  1186          1315  1          ,DSC$K_DTYPE_NZ,DSC$K_DTYPE_P,K_UNSDTYROU,K_UNSDTYROO,K_UNSDESSTA
;  1187          1316  1          ,K_UNSDTYROU,DSC$K_DTYPE_O,DSC$K_DTYPE_G,DSC$K_DTYPE_H,DSC$K_DTYPE_GC
;  1188          1317  1          ,DSC$K_DTYPE_HC,K_UNSDTYROU,K_UNSDTYROO,K_UNSDTYROU,DSC$K_DTYPE_VU
;  1189          1318  1          ,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA,K_UNSDESSTA
;  1190          1319  1          ,DSC$K_DTYPE_TF,DSC$K_DTYPE_SV,DSC$K_DTYPE_SVU
;  1191          1320  1      %( State fourteen.  Class uba. )%
;  1192          1321  1      %( Add more states below )%
;  1193          1322  1      ) : STATES;
;  1194          1323  1
;  1195          1324  1
;  1196          1325  1  ! Final States.
;  1197          1326  1  !
;  1198          1327  1  !These are the final states that are valid CLASS, DATA TYPE combinations.
;  1199          1328  1  !The rest of the final states are error states.
;  1200          1329  1  !The first argument to the macro is CLASS, and the second is the DATA TYPE.
;  1201          1330  1  !
;  1202          1331  1  LITERAL
;  1203          1332  1      K_S_BU = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_BU),
;  1204          1333  1      K_S_WU = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_WU),
;  1205          1334  1      K_S_LU = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_LU),
;  1206          1335  1      K_S_B = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_B),
;  1207          1336  1      K_S_W = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_W),
;  1208          1337  1      K_S_L = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_L),
;  1209          1338  1      K_S_V = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_V),
;  1210          1339  1      K_S_SV = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_SV),
;  1211          1340  1      K_S_TF = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_TF),
;  1212          1341  1      K_S_Q = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_Q),
;  1213          1342  1      K_S_QU = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_QU),
;  1214          1343  1      K_S_O = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_O),
;  1215          1344  1      K_S_F = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_F),
;  1216          1345  1      K_S_FC = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_FC),
;  1217          1346  1      K_S_D = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_D),
;  1218          1347  1      K_S_DC = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_DC),
;  1219          1348  1      K_S_T = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_T),
;  1220          1349  1      K_S_NU = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NU),
;  1221          1350  1      K_S_NL = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NL),
;  1222          1351  1      K_S_NLO = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NLO),
;  1223          1352  1      K_S_NR = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NR),
;  1224          1353  1      K_S_NRO = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NRO),
;  1225          1354  1      K_S_NZ = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_NZ),
;  1226          1355  1      K_S_ZI = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_ZI),
;  1227          1356  1      K_S_P = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_P),
;  1228          1357  1      K_S_G = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_G),
;  1229          1358  1      K_S_GC = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_GC),
;  1230          1359  1      K_S_H = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_H),
;  1231          1360  1      K_S_HC = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_HC),
;  1232          1361  1      K_UBS_V = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_V),
;  1233          1362  1      K_UBS_BU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_BU),
;  1234          1363  1      K_UBS_WU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_WU),
;  1235          1364  1      K_UBS_LU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_LU),
;  1236          1365  1      K_UBS_B = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_B),
;  1237          1366  1      K_UBS_W = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_W),
;  1238          1367  1      K_UBS_L = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_L),
;  1239          1368  1      K_UBS_Q = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_Q),
```

```
;  1240          1369  1     K_UBS_QU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_QU),
;  1241          1370  1     K_UBS_F = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_F),
;  1242          1371  1     K_UBS_D = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_D),
;  1243          1372  1     K_UBS_FC = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_FC),
;  1244          1373  1     K_UBS_DC = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_DC),
;  1245          1374  1     K_UBS_T = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_T),
;  1246          1375  1     K_UBS_NU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NU),
;  1247          1376  1     K_UBS_NL = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NL),
;  1248          1377  1     K_UBS_NLO = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NLO),
;  1249          1378  1     K_UBS_NR = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NR),
;  1250          1379  1     K_UBS_NRO = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NRO),
;  1251          1380  1     K_UBS_NZ = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_NZ),
;  1252          1381  1     K_UBS_P = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_P),
;  1253          1382  1     K_UBS_O = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_O),
;  1254          1383  1     K_UBS_G = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_G),
;  1255          1384  1     K_UBS_H = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_H),
;  1256          1385  1     K_UBS_GC = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_GC),
;  1257          1386  1     K_UBS_HC = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_HC),
;  1258          1387  1     K_UBS_SV = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_SV),
;  1259          1388  1     K_UBS_VU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_VU),
;  1260          1389  1     K_UBS_SVU = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_SVU),
;  1261          1390  1     K_UBS_TF = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_TF),
;  1262          1391  1     K_D_T = FINAL_STATE (DSC$K_CLASS_D, DSC$K_DTYPE_T),
;  1263          1392  1     K_A_BU = FINAL_STATE (DSC$K_CLASS_A, DSC$K_DTYPE_BU),
;  1264          1393  1     K_A_T = FINAL_STATE (DSC$K_CLASS_A, DSC$K_DTYPE_T),
;  1265          1394  1     K_SD_BU = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_BU),
;  1266          1395  1     K_SD_WU = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_WU),
;  1267          1396  1     K_SD_LU = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_LU),
;  1268          1397  1     K_SD_B = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_B),
;  1269          1398  1     K_SD_W = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_W),
;  1270          1399  1     K_SD_L = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_L),
;  1271          1400  1     K_SD_Q = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_Q),
;  1272          1401  1     K_SD_QU = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_QU),
;  1273          1402  1     K_SD_O = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_O),
;  1274          1403  1     K_SD_F = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_F),
;  1275          1404  1     K_SD_FC = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_FC),
;  1276          1405  1     K_SD_D = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_D),
;  1277          1406  1     K_SD_DC = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_DC),
;  1278          1407  1     K_SD_G = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_G),
;  1279          1408  1     K_SD_GC = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_GC),
;  1280          1409  1     K_SD_H = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_H),
;  1281          1410  1     K_SD_HC = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_HC),
;  1282          1411  1     K_SD_T = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_T),
;  1283          1412  1     K_SD_NU = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NU),
;  1284          1413  1     K_SD_NL = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NL),
;  1285          1414  1     K_SD_NLO = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NLO),
;  1286          1415  1     K_SD_NR = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NR),
;  1287          1416  1     K_SD_NRO = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NRO),
;  1288          1417  1     K_SD_NZ = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_NZ),
;  1289          1418  1     K_SD_P = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_P),
;  1290          1419  1     K_NCA_BU = FINAL_STATE (DSC$K_CLASS_NCA, DSC$K_DTYPE_BU),
;  1291          1420  1     K_NCA_T = FINAL_STATE (DSC$K_CLASS_NCA, DSC$K_DTYPE_T),
;  1292          1421  1     K_VS_AC = FINAL_STATE (DSC$K_CLASS_VS, DSC$K_DTYPE_AC),
;  1293          1422  1     K_VS_AZ = FINAL_STATE (DSC$K_CLASS_VS, DSC$K_DTYPE_AZ),
;  1294          1423  1     K_VS_T = FINAL_STATE (DSC$K_CLASS_VS, DSC$K_DTYPE_T),
;  1295          1424  1     K_VS_VT = FINAL_STATE (DSC$K_CLASS_VS, DSC$K_DTYPE_VT),
;  1296          1425  1     K_SMAFINSTA = FINAL_STATE (DSC$K_CLASS_S, DSC$K_DTYPE_V),          ! Smallest final state supported.
```

```
;  1297          1426  1     K_LRGFINSTA = FINAL_STATE (DSC$K_CLASS_UBS, DSC$K_DTYPE_SVU),       ! Largest final state supported.
;  1298          1427  1     K_TOP_SD = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_HC),            ! Top state for class SD.
;  1299          1428  1     K_BOTTOM_SD = FINAL_STATE (DSC$K_CLASS_SD, DSC$K_DTYPE_B);          ! Bottom state for class SD.
```

```
;  1301          1429  1 GLOBAL ROUTINE DBG$COVER_DX_DX (SRC_VALUE_DESC, DST_VALUE_DESC, CVT_ROUND_FLAG) =
;  1302          1430  1 !
;  1303          1431  1 ! FUNCTION
;  1304          1432  1 !       This routine is a cover function for DBG$CVT_DX_DX. It has
;  1305          1433  1 !       two purposes:
;  1306          1434  1 !       1. To declare a handler which screens errors and changes them to
;  1307          1435  1 !       the appropriate DEBUG error.
;  1308          1436  1 !       2. To dummy in the correct class for DBG$CVT_DX_DX
;  1309          1437  1 !
;  1310          1438  1 ! INPUTS
;  1311          1439  1 !       SRC_VALUE_DESC  - Pointer to a value descriptor to be type-converted.
;  1312          1440  1 !
;  1313          1441  1 !       DST_VALUE_DESC  - Pointer to the target value descriptor.
;  1314          1442  1 !
;  1315          1443  1 !       CVT_ROUND_FLAG  - A flag set to TRUE to indicate the rounding takes
;  1316          1444  1 !                         place in conversion.
;  1317          1445  1 !
;  1318          1446  1 ! OUTPUTS
;  1319          1447  1 !       A pointer to a value descriptor is returned.  The target descriptor
;  1320          1448  1 !       is filled in with the result of the conversion.
;  1321          1449  1 !
;  1322          1450  2     BEGIN
;  1323          1451  2
;  1324          1452  2     MAP
;  1325          1453  2         SRC_VALUE_DESC: REF DBG$VALDESC,
;  1326          1454  2         DST_VALUE_DESC: REF DBG$VALDESC;
;  1327          1455  2
;  1328          1456  2     LOCAL
;  1329          1457  2         DUMMY,                          ! A dummy parameter.
;  1330          1458  2         FCODE,                          ! Fcode for the data object
;  1331          1459  2         STATUS,                         ! Return status from typeid check.
;  1332          1460  2         SOURCE_CLASS: BYTE,             ! Class of Source VMS descriptor
;  1333          1461  2         TARGET_CLASS: BYTE,             ! Class of Target VMS descriptor
;  1334          1462  2         SOURCE_DTYPE: BYTE,             ! Dtype of Source VMS descriptor
;  1335          1463  2         TARGET_DTYPE: BYTE,             ! Dtype of Target VMS descriptor
;  1336          1464  2         SOURCE_LENGTH: WORD,            ! Length of Source VMS descriptor
;  1337          1465  2         TARGET_LENGTH: WORD,            ! Length of Target VMS descriptor
;  1338          1466  2         DESC_VAL: REF DBG$VALDESC,      ! Pointer to source or target value descriptor
;  1339          1467  2         DESC_PTR: REF BLOCK[,BYTE],     ! Pointer to source or target descriptor.
;  1340          1468  2         SOURCE: REF BLOCK[,BYTE],       ! Address of VMS descriptor
;  1341          1469  2         TARGET: REF BLOCK[,BYTE],       ! Address of VMS descriptor
;  1342          1470  2         TYPEID_INDEX;                   ! Typeid index to perform the typeid
;  1343          1471  2                                         !       check
;  1344          1472  2
;  1345          1473  2
;  1346          1474  2     ! Recover the VMS descriptors.
;  1347          1475  2     !
;  1348          1476  2     SOURCE = SRC_VALUE_DESC[DBG$A_VALUE_VMSDESC];
;  1349          1477  2     TARGET = DST_VALUE_DESC[DBG$A_VALUE_VMSDESC];
;  1350          1478  2
;  1351          1479  2
;  1352          1480  2     ! Save pointer to result.
;  1353          1481  2     !
;  1354          1482  2     SAVE_RESULT = .DST_VALUE_DESC[DBG$L_VALUE_POINTER];
;  1355          1483  2
;  1356          1484  2
;  1357          1485  2     ! Dummy in the correct class field. (First saving away the old ones.)
```

```
;  1358          1486  2      !
;  1359          1487  2      SOURCE_CLASS = .SOURCE[DSC$B_CLASS];
;  1360          1488  2      TARGET_CLASS = .TARGET[DSC$B_CLASS];
;  1361          1489  2      SOURCE_DTYPE = .SOURCE[DSC$B_DTYPE];
;  1362          1490  2      TARGET_DTYPE = .TARGET[DSC$B_DTYPE];
;  1363          1491  2      SOURCE_LENGTH = .SOURCE[DSC$W_LENGTH];
;  1364          1492  2
;  1365          1493  2
;  1366          1494  2      ! The debugger doesn't handle dynamic string descriptors.  Some output is
;  1367          1495  2      ! better than none, so we treat them as regular string descriptors, and
;  1368          1496  2      ! truncate/pad as required.
;  1369          1497  2      !
;  1370          1498  2      IF .SOURCE_CLASS EQL DSC$K_CLASS_D AND .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_T
;  1371          1499  2      THEN
;  1372          1500  2          SOURCE[DSC$B_CLASS] = DSC$K_CLASS_S;
;  1373          1501  2      IF .TARGET_CLASS EQL DSC$K_CLASS_D AND .TARGET[DSC$B_DTYPE] EQL DSC$K_DTYPE_T
;  1374          1502  2      THEN
;  1375          1503  2          TARGET[DSC$B_CLASS] = DSC$K_CLASS_S;
;  1376          1504  2
;  1377          1505  2
;  1378          1506  2      ! If class field is zero, map in correct class/dtype.
;  1379          1507  2      !
;  1380          1508  2      IF .SOURCE[DSC$B_CLASS] EQL 0
;  1381          1509  2      THEN
;  1382          1510  2          SOURCE[DSC$B_CLASS] = DBG$MAP_DTYPE_CLASS(.SOURCE[DSC$B_DTYPE], FALSE);
;  1383          1511  2
;  1384          1512  2      IF .TARGET[DSC$B_CLASS] EQL 0
;  1385          1513  2      THEN
;  1386          1514  2          TARGET[DSC$B_CLASS] = DBG$MAP_DTYPE_CLASS(.TARGET[DSC$B_DTYPE], FALSE);
;  1387          1515  2
;  1388          1516  2
;  1389          1517  2      ! Case on the fcode.  If the target data is one of the non-standard
;  1390          1518  2      ! data types then typeid and/or range value will be validated by
;  1391          1519  2      ! calling DBG$PERFORM_TYPEID_CHECK.  First set up the routine check
;  1392          1520  2      ! index according to fcode.
;  1393          1521  2      !
;  1394          1522  2      FCODE = .DST_VALUE_DESC[DBG$B_DHDR_FCODE];
;  1395          1523  2      CASE .FCODE FROM RST$K_TYPE_MINIMUM TO RST$K_TYPE_MAXIMUM OF
;  1396          1524  2          SET
;  1397          1525  2          [RST$K_TYPE_ENUM]:
;  1398          1526  2              TYPEID_INDEX = ORT$K_TYPEID_ENUM_ENUM;
;  1399          1527  2
;  1400          1528  2          [RST$K_TYPE_SET]:
;  1401          1529  2              TYPEID_INDEX = ORT$K_TYPEID_SET_SET;
;  1402          1530  2
;  1403          1531  2          [RST$K_TYPE_SUBRNG]:
;  1404          1532  2              TYPEID_INDEX = ORT$K_TYPEID_SUBRNG_SUBRNG;
;  1405          1533  2
;  1406          1534  2          [INRANGE, OUTRANGE]:
;  1407          1535  2              TYPEID_INDEX = 0;
;  1408          1536  2
;  1409          1537  2          TES;
;  1410          1538  2
;  1411          1539  2
;  1412          1540  2      ! If routine check index is set up, call dbg$perform_typeid_check
;  1413          1541  2      ! to perform the typeid check.
;  1414          1542  2      !
```

```
;  1415          1543  2      IF .TYPEID_INDEX NEQ 0
;  1416          1544  2      THEN
;  1417          1545  3          BEGIN
;  1418          1546  3          STATUS = DBG$PERFORM_TYPEID_CHECK(.TYPEID_INDEX,
;  1419          1547  3                          .SRC_VALUE_DESC, .DST_VALUE_DESC, 0);
;  1420          1548  3
;  1421          1549  3          IF NOT .STATUS THEN SIGNAL(DBG$_OPNOTALLOW, 1, .DBG$GL_OPCODE_NAME);
;  1422          1550  2          END;
;  1423          1551  2
;  1424          1552  2
;  1425          1553  2      ! Now, typeid has checked, deposit is legal operation for both
;  1426          1554  2      ! standard and non-standard data types at this point.
;  1427          1555  2      ! Fixup the class and dtype fields to be vax standard format, so
;  1428          1556  2      ! DBG$CVT_DX_DX can be called to perform the conversion.
;  1429          1557  2      !
;  1430          1558  2      INCR I FROM 0 TO 1 DO
;  1431          1559  3          BEGIN
;  1432          1560  3          IF .I EQL 0
;  1433          1561  3          THEN
;  1434          1562  4              BEGIN
;  1435          1563  4              DESC_VAL = .SRC_VALUE_DESC;
;  1436          1564  4              DESC_PTR = .SOURCE;
;  1437          1565  4              END
;  1438          1566  3          ELSE
;  1439          1567  4              BEGIN
;  1440          1568  4              DESC_VAL = .DST_VALUE_DESC;
;  1441          1569  4              DESC_PTR = .TARGET;
;  1442          1570  3              END;
;  1443          1571  3
;  1444          1572  4          IF (.DESC_VAL[DBG$B_VALUE_DTYPE] EQL 0 AND
;  1445          1573  4              .DESC_VAL[DBG$B_VALUE_CLASS] EQL 0)
;  1446          1574  3          THEN
;  1447          1575  3              DESC_PTR = COVER_VMSDESC_SETUP(.DESC_VAL[DBG$L_DHDR_TYPEID],
;  1448          1576  3                                  .DESC_PTR);
;  1449          1577  2          END;
;  1450          1578  2
;  1451          1579  2
;  1452          1580  2      ! Adjust the length of the source.  So we won't get truncation message.
;  1453          1581  2      ! This is used for, ie., DEP enum=1, where enum is allocated 1 byte, and
;  1454          1582  2      ! 1 is 1 longword.  in some cases, we'll get integer overflow message.
;  1455          1583  2      !
;  1456          1584  2      SELECTONE .FCODE OF
;  1457          1585  2          SET
;  1458          1586  3          [RST$K_TYPE_ENUM, RST$K_TYPE_SUBRNG]:
;  1459          1587  3              BEGIN
;  1460          1588  3              IF .SRC_VALUE_DESC[DBG$L_DHDR_TYPEID] EQL 0
;  1461          1589  3              THEN
;  1462          1590  4                  BEGIN
;  1463          1591  4                  IF .SRC_VALUE_DESC[DBG$B_DHDR_FCODE] EQL RST$K_TYPE_ATOMIC
;  1464          1592  4                  THEN
;  1465          1593  5                      BEGIN
;  1466          1594  5                      SOURCE[DSC$B_CLASS] = .TARGET[DSC$B_CLASS];
;  1467          1595  5                      SOURCE[DSC$B_DTYPE] = .TARGET[DSC$B_DTYPE];
;  1468          1596  5                      SOURCE[DSC$W_LENGTH] = .TARGET[DSC$W_LENGTH];
;  1469          1597  4                      END;
;  1470          1598  4
;  1471          1599  3                  END;
```

```
;  1472          1600  3
;  1473          1601  3                END;
;  1474          1602  3
;  1475          1603  3            [OTHERWISE]:
;  1476          1604  3                0;
;  1477          1605  3            TES;
;  1478          1606  3
;  1479          1607  3
;  1480          1608  2        ! Do the conversion.  Put everything back.
;  1481          1609  2        !
;  1482          1610  2        SELECTONE .FCODE OF
;  1483          1611  2            SET
;  1484          1612  2            [RST$K_TYPE_RFA]:
;  1485          1613  2                CH$MOVE(.DST_VALUE_DESC[DBG$W_VALUE_LENGTH],
;  1486          1614  2                        .SRC_VALUE_DESC[DBG$L_VALUE_POINTER], .DST_VALUE_DESC[DBG$L_VALUE_POINTER]);
;  1487          1615  2
;  1488          1616  2            [RST$K_TYPE_SET]:
;  1489          1617  3                BEGIN
;  1490          1618  3                LOCAL
;  1491          1619  3                    INDEX,
;  1492          1620  3                    SETVALUE: REF BITVECTOR[];
;  1493          1621  3
;  1494          1622  3                IF .SRC_VALUE_DESC[DBG$B_DMDR_FCODE] EQL RST$K_TYPE_SET
;  1495          1623  3                THEN
;  1496          1624  4                    BEGIN
;  1497          1625  4                    CH$MOVE(.DST_VALUE_DESC[DBG$W_VALUE_LENGTH],
;  1498          1626  4                            .SRC_VALUE_DESC[DBG$L_VALUE_POINTER], .DST_VALUE_DESC[DBG$L_VALUE_POINTER]);
;  1499          1627  4                    END
;  1500          1628  4
;  1501          1629  3                ELSE
;  1502          1630  4                    BEGIN
;  1503          1631  4                    INDEX = ..SRC_VALUE_DESC[DBG$L_VALUE_POINTER];
;  1504          1632  4                    IF .INDEX LSS 0 THEN SIGNAL(DBG$_BITRANGE);
;  1505          1633  4                    SETVALUE = .DST_VALUE_DESC[DBG$L_VALUE_POINTER];
;  1506          1634  5                    IF .INDEX LEQ (.DST_VALUE_DESC[DBG$W_VALUE_LENGTH] * 8 - 1)
;  1507          1635  4                    THEN
;  1508          1636  4                        SETVALUE[.INDEX] = 1
;  1509          1637  4                    ELSE
;  1510          1638  4                        SIGNAL(DBG$_BITRANGE);
;  1511          1639  3                    END;
;  1512          1640  2                END;
;  1513          1641  2
;  1514          1642  2            [OTHERWISE]:
;  1515          1643  2                DBG$CVT_DX_DX (.SOURCE, .TARGET, DUMMY, .CVT_ROUND_FLAG);
;  1516          1644  2            TES;
;  1517          1645  2
;  1518          1646  2
;  1519          1647  2        SOURCE[DSC$B_CLASS] = .SOURCE_CLASS;
;  1520          1648  2        SOURCE[DSC$B_DTYPE] = .SOURCE_DTYPE;
;  1521          1649  2        SOURCE[DSC$W_LENGTH] = .SOURCE_LENGTH;
;  1522          1650  2        TARGET[DSC$B_CLASS] = .TARGET_CLASS;
;  1523          1651  2        TARGET[DSC$B_DTYPE] = .TARGET_DTYPE;
;  1524          1652  2
;  1525          1653  2
;  1526          1654  2        ! Do range check.
;  1527          1655  2        !
;  1528          1656  2        IF .TYPEID_INDEX NEQ 0
```

```
;  1529          1657  2      THEN
;  1530          1658  3          BEGIN
;  1531          1659  3          STATUS = DBG$PERFORM_TYPEID_CHECK (.TYPEID_INDEX,
;  1532          1660  3                              .SRC_VALUE_DESC, 0, .DST_VALUE_DESC);
;  1533          1661  3          IF NOT .STATUS
;  1534          1662  3          THEN
;  1535          1663  3              SIGNAL (DBG$_IVALOUTBNDS, 1, .DBG$GL_OPCODE_NAME);
;  1536          1664  3          END;
;  1537          1665  2
;  1538          1666  2      RETURN .DST_VALUE_DESC;
;  1539          1667  1      END;


                                                                .TITLE  DBGCVTDX
                                                                .IDENT  \V04-000\

                                                                .PSECT  DBG$PLIT,NOWRT,  SHR,  PIC,0

FF  0D  FF  0B  0A  09  FB  FF  FB  FF  04  FF  02  01  FF  00000 P.AAB:    .BYTE   -1, 1, 2, -1, 4, -1, -5, -1, -5, 9, 10, -
                                                                            11, -1, 13, -1
0E  0D  0C  0B  0A  09  08  07  06  05  04  03  02  01  FE  0000F P.AAC:    .BYTE   -2, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, -
1D  1C  1B  1A  FE  FC  FE  16  15  14  13  12  11  10  0F  0001E                    12, 13, 14, 15, 16, 17, 18, 19, 20, 21, -
FC  FC  FE  29  28  FC  FC  FC  FC  FC  FE  FE  FE  FE  1E  0002D                    22, -2, -4, -2, 26, 27, 28, 29, 30, -2, -
FC  FC  0E  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  0003C                    -2, -2, -2, -4, -4, -4, -4, -4, 40, 41, -
FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  0004B                    -2, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
FD  02  FE  FE  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  0005A                    -4, -4, -4, -4, -4, 14, -4, -4, -4, -4, -4, -
FC  FC  FC  FC  0E  FE  FE  FD  FD  FD  FD  FD  FD  FE  FD  00069                    -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
FC  FC  FE  FC  FE  FD  FD  FE  FE  FE  FC  FC  FC  FC  FC  00078                    -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
05  04  03  02  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  00087                    -4, -4, -4, -4, -2, -2, 2, -3, -3, -2, -
14  13  12  11  10  0F  0E  0D  0C  0B  0A  09  08  07  06  00096                    -3, -3, -3, -3, -3, -3, -2, -2, 14, -4, -
FC  FC  FC  FC  FC  1E  1D  1C  1B  1A  FC  FC  FC  FC  15  000A5                    -4, -4, -4, -4, -4, -4, -4, -4, -2, -2, -
FD  FD  FE  FD  FD  02  FE  FE  FC  FC  FC  FC  FC  FC  FC  000B4                    -2, -3, -3, -2, -4, -2, -4, -4, -4, -4, -
FC  FC  FC  FC  FC  FC  FC  FC  0E  FE  FE  FD  FD  FD  FD  000C3                    -4, -4, -4, -4, -4, -4, -4, -4, -4, 2, 3, -
FC  FC  FC  FC  FC  FC  FE  FC  FE  FD  FD  FE  FE  FE  FC  000D2                    4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, -
FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  000E1                    16, 17, 18, 19, 20, 21, -4, -4, -4, -4, -
FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  0E  FC  FC  FC  FC  000F0                    26, 27, 28, 29, 30, -4, -4, -4, -4, -4, -
27  26  25  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  FC  000FF                    -4, -4, -4, -4, -4, -4, -4, -2, -2, 2, -
08  0A  09  08  07  06  05  04  03  02  01  FE  FC  FC  FC  0010E                    -3, -3, -2, -3, -3, -3, -3, -3, -3, -2, -
1A  FE  FC  FE  FE  15  14  13  12  11  10  0F  0E  0D  0C  0011D                    -2, 14, -4, -4, -4, -4, -4, -4, -4, -4, -
29  28  FC  FC  FC  FC  FC  22  FE  FE  FE  1E  1D  1C  1B  0012C                    -4, -2, -2, -2, -3, -3, -2, -4, -2, -4, -
                                                    2A      0013B                    -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
                                                                                     -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
                                                                                     -4, -4, -4, -4, 14, -4, -4, -4, -4, -4, -
                                                                                     -4, -4, -4, -4, -4, -4, -4, -4, -4, -4, -
                                                                                     -4, -4, -4, -4, -4, -4, -4, 37, 38, 39, -
                                                                                     -4, -4, -4, -2, 1, 2, 3, 4, 5, 6, 7, 8, -
                                                                                     9, 10, 11, 12, 13, 14, 15, 16, 17, 18, -
                                                                                     19, 20, 21, -2, -2, -4, -2, 26, 27, 28, -
                                                                                     29, 30, -2, -2, -2, 34, -4, -4, -4, -4, -
                                                                                     -4, 40, 41, 42

76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0013C P.AAD:    .ASCII  <24>\DBGCVTDX:  invalid class\
                    73  73  61  6C  63  20  64  69  6C  61  0014B
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00155 P.AAE:    .ASCII  <24>\DBGCVTDX:  invalid class\
                    73  73  61  6C  63  20  64  69  6C  61  00164
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0016E P.AAF:    .ASCII  <24>\DBGCVTDX:  invalid class\
                    73  73  61  6C  63  20  64  69  6C  61  0017D
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00187 P.AAG:    .ASCII  <24>\DBGCVTDX:  invalid class\
```

```
                                                73 73 61 6C 63 20 64 69 6C 61  00196
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    001A0 P.AAH:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  001AF
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    001B9 P.AAI:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  001C8
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    001D2 P.AAJ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  001E1
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    001EB P.AAK:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  001FA
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00204 P.AAL:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00213
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    0021D P.AAM:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  0022C
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00236 P.AAN:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00245
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    0024F P.AAO:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  0025E
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00268 P.AAP:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00277
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00281 P.AAQ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00290
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    0029A P.AAR:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  002A9
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    002B3 P.AAS:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  002C2
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    002CC P.AAT:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  002DB
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    002E5 P.AAU:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  002F4
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    002FE P.AAV:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  0030D
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00317 P.AAW:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00326
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00330 P.AAX:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  0033F
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00349 P.AAY:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00358
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00362 P.AAZ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00371
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    0037B P.ABA:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  0038A
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00394 P.ABB:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  003A3
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    003AD P.ABC:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  003BC
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    003C6 P.ABD:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  003D5
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    003DF P.ABE:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  003EE
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    003F8 P.ABF:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00407
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00411 P.ABG:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00420
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    0042A P.ABH:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00439
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18    00443 P.ABI:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                                73 73 61 6C 63 20 64 69 6C 61  00452
```

```
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0045C P.ABJ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00468
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00475 P.ABK:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00484
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0048E P.ABL:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0049D
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 004A7 P.ABM:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 004B6
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 004C0 P.ABN:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 004CF
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 004D9 P.ABO:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 004E8
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 004F2 P.ABP:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00501
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0050B P.ABQ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0051A
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00524 P.ABR:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00533
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0053D P.ABS:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0054C
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00556 P.ABT:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00565
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0056F P.ABU:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0057E
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00588 P.ABV:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00597
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 005A1 P.ABW:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 005B0
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 005BA P.ABX:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 005C9
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 005D3 P.ABY:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 005E2
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 005EC P.ABZ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 005FB
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00605 P.ACA:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00614
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0061E P.ACB:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0062D
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00637 P.ACC:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00646
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00650 P.ACD:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0065F
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00669 P.ACE:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00678
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00682 P.ACF:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 00691
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 0069B P.ACG:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 006AA
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 006B4 P.ACH:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 006C3
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 006CD P.ACI:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 006DC
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 006E6 P.ACJ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 006F5
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 006FF P.ACK:  .ASCII  <24>\DBGCVTDX:  invalid class\
                73 73 61 6C 63 20 64 69 6C 61 0070E
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00718 P.ACL:  .ASCII  <24>\DBGCVTDX:  invalid class\
```

```
                                73  73  61  6C  63  20  64  69  6C  61  00727
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00731 P.ACM:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00740
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0074A P.ACN:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00759
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00763 P.ACO:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00772
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0077C P.ACP:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00788
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00795 P.ACQ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  007A4
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  007AE P.ACR:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  007BD
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  007C7 P.ACS:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  007D6
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  007E0 P.ACT:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  007EF
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  007F9 P.ACU:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00808
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00812 P.ACV:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00821
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0082B P.ACW:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0083A
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00844 P.ACX:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00853
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0085D P.ACY:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0086C
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00876 P.ACZ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00885
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0088F P.ADA:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0089E
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  008A8 P.ADB:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  008B7
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  008C1 P.ADC:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  008D0
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  008DA P.ADD:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  008E9
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  008F3 P.ADE:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00902
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0090C P.ADF:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0091B
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00925 P.ADG:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00934
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  0093E P.ADH:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0094D
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00957 P.ADI:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00966
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00970 P.ADJ:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  0097F
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  00989 P.ADK:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  00998
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  009A2 P.ADL:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  009B1
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  009BB P.ADM:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  009CA
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  009D4 P.ADN:  .ASCII  <24>\DBGCVTDX:  invalid class\
                                73  73  61  6C  63  20  64  69  6C  61  009E3
```

```
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 009ED P.ADO:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 009FC
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A06 P.ADP:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A15
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A1F P.ADQ:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A2E
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A38 P.ADR:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A47
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A51 P.ADS:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A60
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A6A P.ADT:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A79
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A83 P.ADU:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00A92
76 6E 69 20 20 3A 58 44 54 56 43 47 42 44 18 00A9C P.ADV:  .ASCII  <24>\DBGCVTDX:  invalid class\
               73 73 61 6C 63 20 64 69 6C 61 00AAB

                                                          .PSECT  DBG$OWN,NOEXE,  PIC,2

                                                    00000 DECIMAL_CONVERT:
                                                          .BLKB   4
                                                    00004 SAVE_RESULT:
                                                          .BLKB   4

                                                          CLASS_TABLE=        P.AAB
                                                          DTYPE_TABLE=        P.AAC
                                                                  .EXTRN  DBG$CVT_ASHP_R1
                                                                  .EXTRN  DBG$CVT_CMPH_R1
                                                                  .EXTRN  DBG$CVT_CVTDH_R1
                                                                  .EXTRN  DBG$CVT_CVTLB_R1
                                                                  .EXTRN  DBG$CVT_CVTLH_R1
                                                                  .EXTRN  DBG$CVT_CVTLW_R1
                                                                  .EXTRN  DBG$CVT_CVTRDH_R1
                                                                  .EXTRN  DBG$CVT_CVTHD_R1
                                                                  .EXTRN  DBG$CVT_CVTHF_R1
                                                                  .EXTRN  DBG$CVT_CVTHG_R1
                                                                  .EXTRN  DBG$CVT_CVTGH_R1
                                                                  .EXTRN  DBG$CVT_CVTRHL_R1
                                                                  .EXTRN  DBG$CVT_CVTRHO_R1
                                                                  .EXTRN  DBG$CVT_CVTRHQ_R1
                                                                  .EXTRN  DBG$CVT_CVTROUB_R1
                                                                  .EXTRN  DBG$CVT_CVTROUH_R1
                                                                  .EXTRN  DBG$CVT_DIVD2_R1
                                                                  .EXTRN  DBG$CVT_DIVH2_R1
                                                                  .EXTRN  DBG$CVT_DIVP_R1
                                                                  .EXTRN  DBG$CVT_MULD2_R1
                                                                  .EXTRN  DBG$CVT_MULH2_R1
                                                                  .EXTRN  DBG$CVT_MULP_R1
                                                                  .EXTRN  DBG$GET_SET_TYPEID
                                                                  .EXTRN  DBG$INS_ENCODE, DBG$MAP_DTYPE_CLASS
                                                                  .EXTRN  DBG$PERFORM_TYPEID_CHECK
                                                                  .EXTRN  DBG$STA_TYP_SUBRNG
                                                                  .EXTRN  DBG$STA_TYP_ATOMIC
                                                                  .EXTRN  DBG$STRIP_ZEROES
                                                                  .EXTRN  FOR$CVT_D_TE, FOR$CVT_D_TF
                                                                  .EXTRN  FOR$CVT_G_TE, FOR$CVT_G_TF
                                                                  .EXTRN  FOR$CVT_H_TE, FOR$CVT_H_TF
```

```
                                                        .EXTRN  DBG$CVT_SCALE_OU_UP_BY_10_R1
                                                        .EXTRN  DBG$CVT_SCALE_OU_DOWN_BY_TO_R1
                                                        .EXTRN  LIB$$CVT_SCALE_OO_UP_BY_TO_R1
                                                        .EXTRN  LIB$$CVT_SCALE_OU_DOWN_BY_TO_R1
                                                        .EXTRN  DBG$CVT_SCALE_OU_UP_BY_2_R1
                                                        .EXTRN  DBG$CVT_SCALE_OU_DOWN_BY_2_R1
                                                        .EXTRN  LIB$MATCH_COND, LIB$SIG_TO_RET
                                                        .EXTRN  LIB$SCOPY_R_DX6
                                                        .EXTRN  LIB$SCOPY_DXDX6
                                                        .EXTRN  LIB$STOP, MTH$CVT_D_G
                                                        .EXTRN  OTS$CVT_L_TI, OTS$CVT_T_D
                                                        .EXTRN  OTS$CVT_T_G, OTS$CVT_T_R
                                                        .EXTRN  SYS$ASCTIM, SYS$BINTIM
                                                        .EXTRN  LIB$AB_CVTTP_U, LIB$AB_CVT_O_U
                                                        .EXTRN  LIB$AB_CVTTP_O, LIB$AB_CVT_U_O
                                                        .EXTRN  LIB$AB_CVTPT_U, LIB$AB_CVTPT_O
                                                        .EXTRN  LIB$AB_CVTPT_Z, LIB$AB_CVTTP_Z
                                                        .EXTRN  DBG$GL_OPCODE_NAME
                                                        .EXTRN  LIB$_STRTRU

                                                        .PSECT  DBG$CODE,NOWRT,  SHR,  PIC,0

                                        0FFC 00000      .ENTRY  DBG$COVER_DX_DX, Save R2,R3,R4,R5,R6,R7,R8,-;  1429
                                                                R9,R10,R11
                                  5E        24  C2 00002      SUBL2   #36, SP
                                  58    04  AC  D0 00005      MOVL    SRC_VALUE_DESC, R8                      1476
                                  56    14  A8  9E 00009      MOVAB   20(R8), SOURCE
                                  57    08  AC  D0 0000D      MOVL    DST_VALUE_DESC, R7                      1477
                                  59    14  A7  9E 00011      MOVAB   20(R7), TARGET
                       00000000'  EF    18  A7  D0 00015      MOVL    24(R7), SAVE_RESULT                     1482
                                  5B    03  A6  9E 0001D      MOVAB   3(SOURCE), RT1                          1487
                                  10  AE        6B  90 00021  MOVB    (R11), SOURCE_CLASS
                                  5A    03  A9  9E 00025      MOVAB   3(TARGET), R10                          1488
                                  0C  AE        6A  90 00029  MOVB    (R10), TARGET_CLASS
                                  08  AE    02  A6  90 0002D  MOVB    2(SOURCE), SOURCE_DTYPE                 1489
                                  04  AE    02  A9  90 00032  MOVB    2(TARGET), TARGET_DTYPE                 1490
                                      6E        66  B0 00037  MOVW    (SOURCE), SOURCE_LENGTH                 1491
                                      02    10  AE  91 0003A  CMPB    SOURCE_CLASS, #2                        1498
                                                09  12 0003E  BNEQ    1$
                                      0E    02  A6  91 00040  CMPB    2(SOURCE), #14
                                                03  12 00044  BNEQ    1$
                                      6B        01  90 00046  MOVB    #1, (R11)                               1500
                                      02    0C  AE  91 00049  1$:    CMPB    TARGET_CLASS, #2                1501
                                                09  12 0004D  BNEQ    2$
                                      0E    02  A9  91 0004F  CMPB    2(TARGET), #14
                                                03  12 00053  BNEQ    2$
                                      6A        01  90 00055  MOVB    #1, (R10)                               1503
                                      6B        95  00058  2$:    TSTB    (R11)                           1508
                                                10  12 0005A  BNEQ    3$
                                                7E  D4 0005C  CLRL    -(SP)                                   1510
                                      7E    02  A6  9A 0005E  MOVZBL  2(SOURCE), -(SP)
                           00000000G    00        02  FB 00062  CALLS   #2, DBG$MAP_DTYPE_CLASS
                                      6B        50  90 00069  MOVB    R0, (R11)
                                      6A        95  0006C  3$:    TSTB    (R10)                           1512
                                                10  12 0006E  BNEQ    4$
                                                7E  D4 00070  CLRL    -(SP)                                   1514
                                      7E    02  A9  9A 00072  MOVZBL  2(TARGET), -(SP)
```

```
                    00000000G  00          02  FB 00076          CALLS   #2, DBG$MAP_DTYPE_CLASS                          ;
                               6A           50  90 0007D          MOVB    R0, (R10)                                       ;
                               54     06    A7  9A 00080 4$:      MOVZBL  6(R7), FCODE                                    ; 1522
                            15 01           54  CF 00084          CASEL   FCODE, #1, #21                                  ; 1523
                0031    002C       002C         002C  00088 5$:  .WORD   6$-5$,-                                         ;
                0037    002C       002C         002C  00090              6$-5$,-                                         ;
                002C    002C       002C         003D  00098              6$-5$,-                                         ;
                002C    002C       002C         002C  000A0              7$-5$,-                                         ;
                002C    002C       002C         002C  000A8              6$-5$,-                                         ;
                                   002C         002C  000B0              6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         8$-5$,-                                         ;
                                                                         9$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$,-                                         ;
                                                                         6$-5$                                           ;
                                                14     AE   D4 000B4 6$:  CLRL    TYPEID_INDEX                          ; 1535
                                                            10  11 000B7  BRB     10$                                   ;
                                          14    AE          01  D0 000B9 7$:  MOVL    #1, TYPEID_INDEX                      ; 1526
                                                            0A  11 000BD  BRB     10$                                   ;
                                          14    AE          02  D0 000BF 8$:  MOVL    #2, TYPEID_INDEX                      ; 1529
                                                            04  11 000C3  BRB     10$                                   ;
                                          14    AE          04  D0 000C5 9$:  MOVL    #4, TYPEID_INDEX                      ; 1532
                                                   18   AE   D4 000C9 10$:  CLRL    24(SP)                                ; 1543
                                                   14   AE   D5 000CC      TSTL    TYPEID_INDEX                          ;
                                                            30  13 000CF  BEQL    11$                                   ;
                                                   18   AE   D6 000D1      INCL    24(SP)                                ;
                                                            7E  D4 000D4  CLRL    -(SP)                                 ; 1546
                                                            57  DD 000D6  PUSHL   R7                                    ; 1547
                                                            58  DD 000D8  PUSHL   R8                                    ;
                                                   20   AE   DD 000DA      PUSHL   TYPEID_INDEX                          ; 1546
                    00000000G  00               04  FB 000DD          CALLS   #4, DBG$PERFORM_TYPEID_CHECK                ;
                            1C AE               50  D0 000E4          MOVL    R0, STATUS                                  ;
                               15       1C      AE  E8 000E8          BLBS    STATUS, 11$                                 ; 1549
                                   00000000G    00  DD 000EC          PUSHL   DBG$GL_OPCODE_NAME                          ;
                                                01  DD 000F2          PUSHL   #1                                          ;
                                   000289CA     8F  DD 000F4          PUSHL   #166346                                     ;
                    00000000G  00               03  FB 000FA          CALLS   #3, LIB$SIGNAL                              ;
                                                53  D4 00101 11$:     CLRL    I                                           ; 1558
                                                08  12 00103 12$:     BNEQ    13$                                         ; 1560
                               52                   58  D0 00105      MOVL    R8, DESC_VAL                                ; 1563
                               50                   56  D0 00108      MOVL    SOURCE, DESC_PTR                            ; 1564
                                                    06  11 0010B      BRB     14$                                         ; 1560
                               52                   57  D0 0010D 13$:  MOVL    R7, DESC_VAL                                ; 1568
                               50                   59  D0 00110      MOVL    TARGET, DESC_PTR                            ; 1569
                                                   16   A2  95 00113 14$:  TSTB    22(DESC_VAL)                            ; 1572
                                                        0F  12 00116      BNEQ    15$                                     ;
```

```
                                 17  A2  95 00118          TSTB    23(DESC_VAL)                                    ; 1573
                                     0A  12 0011B          BNEQ    15$                                             ;
                                     50  DD 0011D          PUSHL   DESC_PTR                                        ; 1576
                                 08  A2  DD 0011F          PUSHL   8(DESC_VAL)                                     ; 1575
                          0000V  CF      02  FB 00122          CALLS   #2, COVER_VMSDESC_SETUP                         ;
                       D8         53      01  F3 00127 15$:     AOBLEQ  #1, I, 12$                              ; 1558
                                  04      54  D1 0012B          CMPL    FCODE, #4                               ; 1586
                                          05  13 0012E          BEQL    16$                                             ;
                                  09      54  D1 00130          CMPL    FCODE, #9                                       ;
                                          16  12 00133          BNEQ    17$                                             ;
                                 08  A8  D5 00135 16$:     TSTL    8(R8)                                   ; 1588
                                          11  12 00138          BNEQ    17$                                             ;
                                 02  06  A8  91 0013A          CMPB    6(R8), #2                               ; 1591
                                          0B  12 0013E          BNEQ    17$                                             ;
                                  68      6A  90 00140          MOVB    (R10), (R11)                            ; 1594
                             02   A6  02  A9  90 00143          MOVB    2(TARGET), 2(SOURCE)                    ; 1595
                                  66      69  B0 00148          MOVW    (TARGET), (SOURCE)                      ; 1596
                                  14      54  D1 0014B 17$:     CMPL    FCODE, #20                              ; 1612
                                          0B  13 0014E          BEQL    18$                                             ;
                                  08      54  D1 00150          CMPL    FCODE, #8                                       ; 1616
                                          49  12 00153          BNEQ    22$                                             ;
                                  08  06  A8  91 00155          CMPB    6(R8), #8                               ; 1622
                                          09  12 00159          BNEQ    19$                                             ;
                18   B7    18   B8  14  A7  28 0015B 18$:     MOVC3   20(R7), @24(R8), @24(R7)                ; 1626
                                          49  11 00162          BRB     23$                                     ; 1622
                                  52      18  B8  D0 00164 19$:     MOVL    @24(R8), INDEX                          ; 1631
                                          0D  18 00168          BGEQ    20$                                     ; 1632
                                 00028248 8F  DD 0016A          PUSHL   #164424                                         ;
                       00000000G 00      01  FB 00170          CALLS   #1, LIB$SIGNAL                                  ;
                                  51  18  A7  D0 00177 20$:     MOVL    24(R7), SETVALUE                        ; 1633
                                  50  14  A7  3C 0017B          MOVZWL  20(R7), R0                              ; 1634
                                  50      08  C4  0017F          MULL2   #8, R0                                          ;
                                          50  D7 00182          DECL    R0                                              ;
                                  50      52  D1 00184          CMPL    INDEX, R0                                       ;
                                          06  14 00187          BGTR    21$                                             ;
                20                61      52  E2 00189          BBSS    INDEX, (SETVALUE), 23$                  ; 1636
                                          1E  11 0018D          BRB     23$                                             ;
                                 00028248 8F  DD 0018F 21$:     PUSHL   #164424                                 ; 1638
                       00000000G 00      01  FB 00195          CALLS   #1, LIB$SIGNAL                                  ;
                                          0F  11 0019C          BRB     23$                                     ; 1610
                                 0C  AC  DD 0019E 22$:     PUSHL   CVT_ROUND_FLAG                          ; 1643
                                 24  AE  9F 001A1          PUSHAB  DUMMY                                           ;
                                0240 8F  BB 001A4          PUSHR   #^M<R6,R9>                                      ;
                          0000V  CF      04  FB 001A8          CALLS   #4, DBG$CVT_DX_DX                               ;
                                  6B  10  AE  90 001AD 23$:     MOVB    SOURCE_CLASS, (R11)                     ; 1647
                             02   A6  08  AE  90 001B1          MOVB    SOURCE_DTYPE, 2(SOURCE)                 ; 1648
                                  66      6E  B0 001B6          MOVW    SOURCE_LENGTH, (SOURCE)                 ; 1649
                                  6A  0C  AE  90 001B9          MOVB    TARGET_CLASS, (R10)                     ; 1650
                             02   A9  04  AE  90 001BD          MOVB    TARGET_DTYPE, 2(TARGET)                 ; 1651
                                  2D  18  AE  E9 001C2          BLBC    24(SP), 24$                             ; 1656
                                          57  DD 001C6          PUSHL   R7                                      ; 1660
                                          7E  D4 001C8          CLRL    -(SP)                                   ; 1659
                                          58  DD 001CA          PUSHL   R8                                      ; 1660
                                 20  AE  DD 001CC          PUSHL   TYPEID_INDEX                            ; 1659
                       00000000G 00      04  FB 001CF          CALLS   #4, DBG$PERFORM_TYPEID_CHECK                    ;
                                1C  AE  50  D0 001D6          MOVL    R0, STATUS                                      ;
                                  15  1C  AE  E8 001DA          BLBS    STATUS, 24$                             ; 1661
```

```
                                00000000G  00  DD 001DE          PUSHL   DBG$GL_OPCODE_NAME                       ; 1663
                                           01  DD 001E4          PUSHL   #1                                      ;
                                0002874B   8F  DD 001E6          PUSHL   #165707                                 ;
                     00000000G  00          03  FB 001EC          CALLS   #3, LIB$SIGNAL                          ;
                                50          57  D0 001F3 248:     MOVL    R7, R0                                  ; 1666
                                            04  001F6           RET                                             ; 1667

; Routine Size:  503 bytes,    Routine Base:  DBG$CODE + 0000
```

```
;  1541          1668  1 ROUTINE COVER_VMSDESC_SETUP(TYPEID, VMSDESC) =
;  1542          1669  1 !
;  1543          1670  1 ! FUNCTION
;  1544          1671  1 !       This routine is a hack routine called depending on FCODE in the
;  1545          1672  1 !       TYPEID. The purpose of this routine is to plunge in the class code
;  1546          1673  1 !       and dtype so that the DBG$CVT_DX_DX can be called.
;  1547          1674  1 !
;  1548          1675  1 ! INPUTS
;  1549          1676  1 !       TYPEID  - Typeid of the data object.
;  1550          1677  1 !
;  1551          1678  1 !       VMSDESC - Vax standard Descriptor.
;  1552          1679  1 !
;  1553          1680  1 ! OUTPUTS
;  1554          1681  1 !       VMSDESC is returned.
;  1555          1682  1 !
;  1556          1683  1
;  1557          1684  2     BEGIN
;  1558          1685  2     MAP
;  1559          1686  2         TYPEID: REF RST$ENTRY,
;  1560          1687  2         VMSDESC: REF BLOCK[,BYTE];
;  1561          1688  2
;  1562          1689  2     VMSDESC[DSC$B_CLASS] = DSC$K_CLASS_S;
;  1563          1690  2     SELECTONE .TYPEID[RST$B_FCODE] OF
;  1564          1691  2         SET
;  1565          1692  2         [RST$K_TYPE_ENUM]:
;  1566          1693  3             BEGIN
;  1567          1694  3             SELECTONE .VMSDESC[DSC$W_LENGTH] OF
;  1568          1695  3                 SET
;  1569          1696  3                 [1]:
;  1570          1697  3                     VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_BU;
;  1571          1698  3                 [2]:
;  1572          1699  3                     VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_WU;
;  1573          1700  3                 [OTHERWISE]:
;  1574          1701  3                     VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_LU;
;  1575          1702  3                 TES;
;  1576          1703  2             END;
;  1577          1704  2
;  1578          1705  2         [RST$K_TYPE_SUBRNG]:
;  1579          1706  3             BEGIN
;  1580          1707  3             LOCAL
;  1581          1708  3                 DUMMY1, DUMMY2, DUMMY3, DTYPE;
;  1582          1709  3
;  1583          1710  3             WHILE .TYPEID[RST$B_FCODE] EQL RST$K_TYPE_SUBRNG DO
;  1584          1711  3                 DBG$STA_TYP_SUBRNG(.TYPEID, TYPEID, DUMMY1, DUMMY2, DUMMY3);
;  1585          1712  3
;  1586          1713  3             SELECTONE .TYPEID[RST$B_FCODE] OF
;  1587          1714  3                 SET
;  1588          1715  3                 [RST$K_TYPE_ENUM]:
;  1589          1716  3                     VMSDESC = COVER_VMSDESC_SETUP(.TYPEID, .VMSDESC);
;  1590          1717  3
;  1591          1718  3                 [RST$K_TYPE_ATOMIC]:
;  1592          1719  4                     BEGIN
;  1593          1720  4                     DBG$STA_TYP_ATOMIC(.TYPEID, DTYPE, DUMMY3);
;  1594          1721  4                     IF .DTYPE EQL DST$K_BOOL THEN DTYPE = DSC$K_DTYPE_TF;
;  1595          1722  4                     VMSDESC[DSC$B_DTYPE] = .DTYPE;
;  1596          1723  4
;  1597          1724  4
```

```
;  1598          1725  4                    ! In here we have a bit of problem, for the size of
;  1599          1726  4                    ! the parant and the size of the subrange is different.
;  1600          1727  4                    ! For example, subrange's parant can be longword integer
;  1601          1728  4                    ! and subrange can be 1 byte.  In order for the type
;  1602          1729  4                    ! converter to take the right value, we adjust the dtype
;  1603          1730  4                    ! by the length.
;  1604          1731  4                    !
;  1605          1732  5                    IF (.DTYPE NEQ DSC$K_DTYPE_T AND .DTYPE NEQ DSC$K_DTYPE_TF)
;  1606          1733  4                    THEN
;  1607          1734  5                        BEGIN
;  1608          1735  5                        SELECTONE .VMSDESC[DSC$W_LENGTH] OF
;  1609          1736  5                            SET
;  1610          1737  5                            [1]:
;  1611          1738  5                                VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_B;
;  1612          1739  5                            [2]:
;  1613          1740  5                                VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_W;
;  1614          1741  5                            [OTHERWISE]:
;  1615          1742  5                                VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_L;
;  1616          1743  5                            TES;
;  1617          1744  5                        END;
;  1618          1745  4                    END;
;  1619          1746  4
;  1620          1747  3                TES;
;  1621          1748  3
;  1622          1749  2            END;
;  1623          1750  2
;  1624          1751  2            [RST$K_TYPE_SET, RST$K_TYPE_TPTR]:
;  1625          1752  2                VMSDESC[DSC$B_DTYPE] = DSC$K_DTYPE_L;
;  1626          1753  2
;  1627          1754  2            [RST$K_TYPE_RFA]:
;  1628          1755  2                VMSDESC[DSC$B_CLASS] = 0;
;  1629          1756  2
;  1630          1757  2            [OTHERWISE]:
;  1631          1758  2                $DBG_ERROR('DBGCVTDX\COVER_VMSDESC_SETUP');
;  1632          1759  2            TES;
;  1633          1760  2
;  1634          1761  2        RETURN .VMSDESC;
;  1635          1762  1        END;


                                                                .PSECT  DBG$PLIT,NOWRT,  SHR,  PIC,0

52 45 56 4F 43 5C 58 44 54 56 43 47 42 44 1C 00AB5 P.ADW:    .ASCII  <28>\DBGCVTDX\<92>\COVER_VMSDESC_SETUP\      ;
   50 55 54 45 53 5F 43 53 45 44 53 4D 56 5F 00AC4                                                           ;


                                                                .PSECT  DBG$CODE,NOWRT,  SHR,  PIC,0

                                                  000C 00000 COVER_VMSDESC_SETUP:
                                                                .WORD   Save R2,R3                          ; 1668
                                        5E          10  C2 00002        SUBL2   #16, SP                     ;
                                        52      08  AC  D0 00005        MOVL    VMSDESC, R2                 ; 1689
                                    03  A2          01  90 00009        MOVB    #1, 3(R2)                   ;
                                        50      04  AC  D0 0000D        MOVL    TYPEID, R0                  ; 1690
                                        50          18  C0 00011        ADDL2   #24, R0                     ;
```

```
                        50         60  9A 00014          MOVZBL  (R0), R0
                        04         50  91 00017          CMPB    R0, #4                          ;1692
                                   1C  12 0001A          BNEQ    3$
                        01         62  B1 0001C          CMPW    (R2), #1                        ;1696
                                   06  12 0001F          BNEQ    1$
                    02  A2         02  90 00021          MOVB    #2, 2(R2)                       ;1697
                                   53  11 00025          BRB     6$
                        02         62  B1 00027  1$:     CMPW    (R2), #2                        ;1698
                                   06  12 0002A          BNEQ    2$
                    02  A2         03  90 0002C          MOVB    #3, 2(R2)                       ;1699
                                   48  11 00030          BRB     6$
                    02  A2         04  90 00032  2$:     MOVB    #4, 2(R2)                       ;1701
                                   42  11 00036          BRB     6$                              ;1690
                        09         50  91 00038  3$:     CMPB    R0, #9                          ;1705
                                   03  13 0003B          BEQL    4$
                                0085  31 0003D          BRW     10$
                        50  04     AC  D0 00040  4$:     MOVL    TYPEID, R0                      ;1710
                        09  18     A0  91 00044          CMPB    24(R0), #9
                                   17  12 00048          BNEQ    5$
                            08     AE  9F 0004A          PUSHAB  DUMMY3                          ;1711
                            04     AE  9F 0004D          PUSHAB  DUMMY2
                            0C     AE  9F 00050          PUSHAB  DUMMY1
                            04     AC  9F 00053          PUSHAB  TYPEID
                                   50  DD 00056          PUSHL   R0
          00000000G         00       05  FB 00058          CALLS   #5, DBG$STA_TYP_SUBRNG
                                   DF  11 0005F          BRB     4$
                        53  04     AC  D0 00061  5$:     MOVL    TYPEID, R3                      ;1713
                        50  18     A3  9A 00065          MOVZBL  24(R3), R0
                        04         50  91 00069          CMPB    R0, #4                          ;1715
                                   0E  12 0006C          BNEQ    7$
                                   52  DD 0006E          PUSHL   R2                              ;1716
                                   53  DD 00070          PUSHL   R3
                    8A  AF         02  FB 00072          CALLS   #2, COVER_VMSDESC_SETUP
                    08  AC         50  D0 00076          MOVL    R0, VMSDESC
                                   78  11 0007A  6$:     BRB     14$
                        02         50  91 0007C  7$:     CMPB    R0, #2                          ;1718
                                   73  12 0007F          BNEQ    14$
                            08     AE  9F 00081          PUSHAB  DUMMY3                          ;1720
                            10     AE  9F 00084          PUSHAB  DTYPE
                                   53  DD 00087          PUSHL   R3
          00000000G         00       03  FB 00089          CALLS   #3, DBG$STA_TYP_ATOMIC
          0000009E          8F     0C  AE  D1 00090          CMPL    DTYPE, #158                     ;1721
                                   04  12 00098          BNEQ    8$
                        0C  AE         28  D0 0009A          MOVL    #40, DTYPE
                    02  A2     0C  AE  90 0009E  8$:     MOVB    DTYPE, 2(R2)                    ;1722
                        0E     0C  AE  D1 000A3          CMPL    DTYPE, #14                      ;1732
                                   4B  13 000A7          BEQL    14$
                        28     0C  AE  D1 000A9          CMPL    DTYPE, #40
                                   45  13 000AD          BEQL    14$
                        01         62  B1 000AF          CMPW    (R2), #1                        ;1737
                                   06  12 000B2          BNEQ    9$
                    02  A2         06  90 000B4          MOVB    #6, 2(R2)                       ;1738
                                   3A  11 000B8          BRB     14$
                        02         62  B1 000BA  9$:     CMPW    (R2), #2                        ;1739
                                   10  12 000BD          BNEQ    11$
                    02  A2         07  90 000BF          MOVB    #7, 2(R2)                       ;1740
                                   2F  11 000C3          BRB     14$
```

```
                         06              50  91 000C5 10$:    CMPB    R0, #6                          ; 1751
                                         05  13 000C8         BEQL    11$
                         08              50  91 000CA         CMPB    R0, #8                          ;
                                         06  12 000CD         BNEQ    12$                             ;
                   02    A2              08  90 000CF 11$:    MOVB    #8, 2(R2)                       ; 1752
                                         1F  11 000D3         BRB     14$                             ;
                         14              50  91 000D5 12$:    CMPB    R0, #20                         ; 1754
                                         05  12 000D8         BNEQ    13$                             ;
                               03        A2  94 000DA         CLRB    3(R2)                           ; 1755
                                         15  11 000DD         BRB     14$                             ;
                         00000000'       EF  9F 000DF 13$:    PUSHAB  P.ADW                           ; 1758
                                         01  DD 000E5         PUSHL   #1                              ;
                         00028362        8F  DD 000E7         PUSHL   #164706                         ;
                00000000G  00            03  FB 000ED         CALLS   #3, LIB$SIGNAL                  ;
                         50        08    AC  D0 000F4 14$:    MOVL    VMSDESC, R0                     ; 1761
                                         04  000F8         RET                                     ; 1762

; Routine Size:  249 bytes,    Routine Base:  DBG$CODE + 01F7
```

```
;  1637          1763  1 GLOBAL ROUTINE DBG$CVT_DX_DX (SOURCE, DESTINATION, OUTLEN):  NOVALUE =
;  1638          1764  1 !
;  1639          1765  1 !                   This is the general data type conversion facility.
;  1640          1766  1 !                   Given two parameters, one the source descriptor,
;  1641          1767  1 !                   second the destination descriptor this routine
;  1642          1768  1 !                   will convert the source to destination.
;  1643          1769  1 !                   The permitted set of class, data type and combination
;  1644          1770  1 !                   of the two is a subset of the ones allowed in the
;  1645          1771  1 !                   calling standard.
;  1646          1772  1 !
;  1647          1773  1 !           The following is a general description of DBG$CVT_DX_DX.
;  1648          1774  1 !
;  1649          1775  1 !           This module is divided into two routines on the bases of functional
;  1650          1776  1 !           modularity.  The front-end (FIND_CVT_PATH), and back-end (DBG$CVT_DX_DX).
;  1651          1777  1 !           The front-end converts the source into an intermediate data type, and
;  1652          1778  1 !           frees the back-end of any error checking of invalid classes and/or
;  1653          1779  1 !           data types (or combination of the two), and of decisions that require
;  1654          1780  1 !           knowledge of which class or data type is being converted.  The only
;  1655          1781  1 !           information that the back-end knows about is what the conversion path
;  1656          1782  1 !           is, and where the intermediate data is.  The back-end then scales the
;  1657          1783  1 !           intermediate data if necessary and converts it to the destination
;  1658          1784  1 !           data type.  Note that even though a scale may not be necessary, the
;  1659          1785  1 !           intermediate data is still converted to a second intermediate data type
;  1660          1786  1 !           just to be consistent.
;  1661          1787  1 !
;  1662          1788  1 !       1.  Upon entry to DBG$CVT_DX_DX, FIND_CVT_PATH routine is called.
;  1663          1789  1 !           FIND_CVT_PATH has 4 functions, they are:
;  1664          1790  1 !                   a. Find any errors concerning the class and data type of
;  1665          1791  1 !                      source and destination descriptor.  These errors can be
;  1666          1792  1 !                      invalid class, invalid data type, or invalid combination
;  1667          1793  1 !                      of a class and data type.  It can also tell which descriptors
;  1668          1794  1 !                      are supported by the VAX-11 calling standard and which are
;  1669          1795  1 !                      supported by this routine.
;  1670          1796  1 !
;  1671          1797  1 !                   b. Figure out what the conversion path is, i.e. class,dtype --> class,dtype.
;  1672          1798  1 !                      These paths are given names such as K_SMLINT_DEC, which reads
;  1673          1799  1 !                      "from small integer to decimal" (categories are defined later).
;  1674          1800  1 !
;  1675          1801  1 !                   c. Convert the source data to an intermediate data.  The strategy
;  1676          1802  1 !                      used to select the appropriate intermediate data is explained later.
;  1677          1803  1 !                      Precision should not be lost in converting to the intermediate type.
;  1678          1804  1 !
;  1679          1805  1 !                   d. Put whatever information needed about the source and destination
;  1680          1806  1 !                      descriptor in two structures passed by DBG$CVT_DX_DX.
;  1681          1807  1 !                      These two structures SRC_INFO, and DST_INFO, contain the kind
;  1682          1808  1 !                      of information that can only be visible when the class, and
;  1683          1809  1 !                      data type of the source and destination descriptors are being
;  1684          1810  1 !                      manipulated. These two structures can be expanded to contain
;  1685          1811  1 !                      more information as new class, and data types may require it.
;  1686          1812  1 !
;  1687          1813  1 !
;  1688          1814  1 !       2.  The following is an overview of the design of FIND_CVT_PATH:
;  1689          1815  1 !           The problem to be solved is to recognize "valid" descriptors.
;  1690          1816  1 !           A descriptor is valid if the CLASS and DATA TYPE fields of the
;  1691          1817  1 !           descriptor satisfy certain conditions.
;  1692          1818  1 !           With this problem in mind we shall use some formal language theory
;  1693          1819  1 !           and applications to solve it.
```

```
;1694    1820  1 !        Let us take a hypothetical problem that is very close but smaller in
;1695    1821  1 !        magnitude of the original problem and solve it.
;1696    1822  1 !        Suppose that the set of classes that we are interested in are
;1697    1823  1 !        CLASS = { c1, c2, c3 }, and the set of data types are
;1698    1824  1 !        DTYPE = { d1, d2, d3, d4 }.  Then suppose that only a certain combinations
;1699    1825  1 !        of CLASS and DTYPE are valid, and they are c1d3, c2d1, c3d2, c3d4.
;1700    1826  1 !        Hence language L(G) is consisted of sentences { c1d3, c2d1, c3d2, c3d4 }.
;1701    1827  1 !        First we need to come up with a grammar for the language L(G).
;1702    1828  1 !        Grammar for L(G) :
;1703    1829  1 !              Z  -->  <S1>d3 | <S2>d1 | <S3>d2 | <S3>d4
;1704    1830  1 !              S1 -->  c1
;1705    1831  1 !              S2 -->  c2
;1706    1832  1 !              S3 -->  c3
;1707    1833  1 !              S4 -->  c4
;1708    1834  1 !        A close look shows that this is a Chomsky type 3 regular grammar,
;1709    1835  1 !        because productions are all
;1710    1836  1 !              NON-TERMINAL --> terminal
;1711    1837  1 !                    or
;1712    1838  1 !              NON_TERMINAL --> <NON-TERMINAL>terminal
;1713    1839  1 !        This type of grammar has the nice feature that its sentential forms
;1714    1840  1 !        can be "accepted" by a finite state machine.
;1715    1841  1 !        The sentential forms of this grammar can also be accepted by a
;1716    1842  1 !        deterministic finite automaton (DFA) because each right hand side
;1717    1843  1 !        has a unique left hand side.
;1718    1844  1 !        A DFA can be written to recognize sentences of this grammar and to
;1719    1845  1 !        reject sentences that are not in the language.
;1720    1846  1 !        The original problem is very similar to this hypothetical one, the
;1721    1847  1 !        only difference is that the set of CLASSES and DTYPES is larger.
;1722    1848  1 !        FIND_CVT_PATH is just a DFA that accepts sentences of language L(V)
;1723    1849  1 !        when L(V) is pairs of VAX-11 DSC$K_CLASS_x DSC$K_DTYPE_y.  The
;1724    1850  1 !        grammar for L(V) is very similar to the grammar for L(G) above.
;1725    1851  1 !
;1726    1852  1 !     3. In order to achieve the conflicting goals:  fast, not large in size,
;1727    1853  1 !        expandable, no loss of precision as a result of intermediate values,
;1728    1854  1 !        there is a need for  a  compromise.  The strategy for categorizing
;1729    1855  1 !        the data types is based on three goals:  precision should not be lost
;1730    1856  1 !        as a result of converting to intermediate data types, data types of
;1731    1857  1 !        the same category should share similar internal representations so
;1732    1858  1 !        they can be converted to and from each other easily, and data types
;1733    1859  1 !        that have to be converted through software should be separated from
;1734    1860  1 !        those that have associated machine instructions.  The third goal
;1735    1861  1 !        provides easy and fast conversions for those data types with
;1736    1862  1 !        associated machine instructions.
;1737    1863  1 !        The current categories were formulated by the following strategy:
;1738    1864  1 !            Divide  the  integers into two groups, small and large integers.
;1739    1865  1 !            Divide the floating numbers into two categories small and large
;1740    1866  1 !            floating. The small category will be the data types
;1741    1867  1 !            that machine instructions are available for their conversions.
;1742    1868  1 !            The large category consist of data types that there are no machine instructions for
;1743    1869  1 !            their conversions or the instructions must be emulated (LIB$EMULATE)
;1744    1870  1 !            for some VAX machines.
;1745    1871  1 !        This categorization will provide conversions that are fast and smooth.
;1746    1872  1 !        As a result we have the following :
;1747    1873  1 !
;1748    1874  1 !        INTEGER       --> SMALL_INTEGER | LARGE_INTEGER
;1749    1875  1 !        FLOAT         --> SMALL_FLOAT | LARGE_FLOAT
;1750    1876  1 !        SMALL_INTEGER --> bu | wu | b | w | l          !Intermediate L
```

```
;  1751          1877  1 !          LARGE_INTEGER --> lu ! q                          !Intermediate OU
;  1752          1878  1 !          SMALL_FLOAT   --> f ! d                           !Intermediate D
;  1753          1879  1 !          LARGE_FLOAT   --> g ! h                           !Intermediate H
;  1754          1880  1 !          DEC           --> nu ! nl ! nlo ! nr ! nro ! nz !Intermediate P
;  1755          1881  1 !          NBDS          --> nbds                            !Intermediate T
;  1756          1882  1 !
;  1757          1883  1 !      4.  Upon return from FIND_CVT_PATH, the main routine then enters a
;  1758          1884  1 !          CASE statement that selects the desired conversion.  This CASE
;  1759          1885  1 !          is explained in detail in the first paragraph of the statement.
;  1760          1886  1 !
;  1761          1887  1 ! AUTHOR:       Farokh Morshed          01-09-1981
;  1762          1888  1 !
;  1763          1889  1 ! FUNCTIONAL DESCRIPTION:
;  1764          1890  1 !
;  1765          1891  1 !       Upon entry, FIND_CVT_PATH is called to identify which conversion is to be
;  1766          1892  1 !       done, i.e. from which CLASS, DTYPE combination to which CLASS, DTYPE
;  1767          1893  1 !       combination.
;  1768          1894  1 !       Also, FIND_CVT_PATH will do all the work of identifying the errors such
;  1769          1895  1 !       as unsupported class, data type, or combinations.
;  1770          1896  1 !       This routine is just a tree of CASE statements, where the outermost
;  1771          1897  1 !       level CASE statement labels have been determined by the FIND_CVT_PATH
;  1772          1898  1 !       routine.
;  1773          1899  1 !
;  1774          1900  1 !
;  1775          1901  1 ! CALLING SEQUENCE:
;  1776          1902  1 !
;  1777          1903  1 !       ret_status.wlc.v = DBG$CVT_DX_DX ( SOURCE.rx.dx, DESTINATION.wx.dx
;  1778          1904  1 !                                          <OUTLEN.wwu.r> <CVT_ROUND_FLAG>)
;  1779          1905  1 !
;  1780          1906  1 ! FORMAL PARAMETERS:
;  1781          1907  1 !
;  1782          1908  1 !       SOURCE                  Address of source descriptor.
;  1783          1909  1 !       DESTINATION             Address of destination descriptor.
;  1784          1910  1 !       OUTLEN                  Output length.  Optional parameter for this
;  1785          1911  1 !                               routine to put the length of actual data (without padding) in.
;  1786          1912  1 !                               This is used only when destination is of data
;  1787          1913  1 !                               type T.
;  1788          1914  1 !       CVT_ROUND_FLAG          A flag set to true to indicate the conversion
;  1789          1915  1 !                               result is rounded.
;  1790          1916  1 !
;  1791          1917  1 ! IMPLICIT INPUTS:
;  1792          1918  1 !
;  1793          1919  1 !       NONE
;  1794          1920  1 !
;  1795          1921  1 ! IMPLICIT OUTPUTS:
;  1796          1922  1 !
;  1797          1923  1 !       NONE
;  1798          1924  1 !
;  1799          1925  1 ! SIDE EFFECTS
;  1800          1926  1 !       Every routine in this module turns on every arithmetic trap in PSW.
;  1801          1927  1 !       Caller must have LIB$EMULATE as handler if any G or H conversions are
;  1802          1928  1 !       asked for.
;  1803          1929  1 !
;  1804          1930  2     BEGIN
;  1805          1931  2
;  1806          1932  2     LOCAL
;  1807          1933  2         CVT_ROUND_FLAG,
```

```
;  1808          1934  2          SRC_INFO:  BLOCK [K_SRC_INFO_LENGTH, BYTE] FIELD (SRC_INFO_FIELDS),     ! Source information structu
;  1809          1935  2          DST_INFO:  BLOCK [K_DST_INFO_LENGTH, BYTE] FIELD (DST_INFO_FIELDS),     ! Destination information st
;  1810          1936  2          OUTPUT_BUFFER:  BLOCK[K_OUTPUT_BUFFER_LENGTH, BYTE],
;  1811          1937  2          OUTPUT,
;  1812          1938  2          DESTINATION_PTR,                                                        ! Ptr to destination
;  1813          1939  2          INTMED_DATA:  BLOCK [K_INTMED_DATA_LENGTH, BYTE],                       ! Intermediate data buffer.
;  1814          1940  2          TEMP_BUF1:  BLOCK [K_TEMP_BUF_LENGTH, BYTE],                            ! Holds temporary data.
;  1815          1941  2          TEMP_BUF2:  BLOCK [K_TEMP_BUF_LENGTH, BYTE],                            ! Holds temporary data.
;  1816          1942  2          CLASS_S_DESC:  BLOCK [8, BYTE],                                         ! A class S descriptor for a
;  1817          1943  2          CLASS_SOURCE:  BYTE,                                                    ! Class of source VMS descri
;  1818          1944  2          CLASS_TARGET:  BYTE,                                                    ! Class of target VMS descri
;  1819          1945  2          FINAL_LEN,                                                              ! Length of data in TEMP_BUF
;  1820          1946  2          CVT_PATH,                                                               ! Calculated convert path.
;  1821          1947  2          NO_DIGITS,                                                              ! Number of digits in decima
;  1822          1948  2          DIGITS_IN_FRACT,                                                        ! Fractional digits (OTS$CVT
;  1823          1949  2          FLOAT_SCALE,                                                            ! Scale of a floating number
;  1824          1950  2          SIGN,                                                                   ! Sign of the floating numbe
;  1825          1951  2          STATUS,                                                                 ! Return status.
;  1826          1952  2          LRGST_P_LU,                                                             ! Largest LU in a packed dec
;  1827          1953  2          LRGST_D_LU,                                                             ! Largest LU in a double flo
;  1828          1954  2          PACK_ZERO,                                                              ! A zero in a packed decimal
;  1829          1955  2          LRGST_H_LU,                                                             ! Largest LU in a H floating
;  1830          1956  2          BUF_OFFSET,                                                             ! Offset to actual data in T
;  1831          1957  2          NEXT_BLANK,                                                             ! Location of next blank in
;  1832          1958  2          OUTPUT_STR_LEN,                                                         ! Length of actual string th
;  1833          1959  2          SRC_POS,                                                                ! Bit offset in source.
;  1834          1960  2          DST_POS,                                                                ! Bit offset in destination.
;  1835          1961  2                                                                                  ! to destination (optional p
;  1836          1962  2          BIN_SCALE,
;  1837          1963  2          SCALE;                                                                  ! Effective scale (source sc
;  1838          1964  2
;  1839          1965  2      MAP
;  1840          1966  2          OUTPUT:  REF BLOCK[, BYTE],
;  1841          1967  2          DESTINATION_PTR: REF BLOCK[, BYTE],
;  1842          1968  2          SOURCE:  REF BLOCK[,BYTE],                                              ! Source VMS descriptor.
;  1843          1969  2          DESTINATION:  REF BLOCK[,BYTE];                                         ! Destination VMS descriptor
;  1844          1970  2
;  1845          1971  2      BUILTIN
;  1846          1972  2          ACTUALPARAMETER,
;  1847          1973  2          ACTUALCOUNT;
;  1848          1974  2
;  1849          1975  2
;  1850          1976  2      ! Establish CVT_HANDLER as handler.
;  1851          1977  2      !
;  1852          1978  2      ENABLE
;  1853          1979  2          CVT_HANDLER;
;  1854          1980  2
;  1855          1981  2
;  1856          1982  2      ! These literals are used a few lines down to test whether we
;  1857          1983  2      ! are doing a conversion from decimal string or packed.
;  1858          1984  2      !
;  1859          1985  2      LITERAL
;  1860          1986  2          MIN_DEC_DTYPE = DSC$K_DTYPE_NU, ! 15
;  1861          1987  2          MAX_DEC_DTYPE = DSC$K_DTYPE_P;  ! 21
;  1862          1988  2
;  1863          1989  2      !++
;  1864          1990  2      !  If the destination or source is Absolute Date Time cut it off here
```

```
;  1865          1991  2     !  and do the conversion.
;  1866          1992  2     !--
;  1867          1993  2     IF (.DESTINATION [DSC$B_DTYPE] EQL DSC$K_DTYPE_ADT)  OR
;  1868          1994  2        (.SOURCE [DSC$B_DTYPE] EQL DSC$K_DTYPE_ADT)
;  1869          1995  2     THEN
;  1870          1996  2         IF (.DESTINATION [DSC$B_DTYPE] EQL DSC$K_DTYPE_T)  AND                    ! ADT to text
;  1871          1997  2            (.SOURCE [DSC$B_DTYPE] EQL DSC$K_DTYPE_ADT)
;  1872          1998  2         THEN
;  1873          1999  2             BEGIN
;  1874          2000  2             LOCAL
;  1875          2001  2                 TEMP;
;  1876          2002  2             CLASS_S_DESC[dsc$b_class]   = dsc$k_class_s;                          ! Build the string descripto
;  1877          2003  2             CLASS_S_DESC[dsc$b_dtype]   = dsc$k_dtype_t;
;  1878          2004  2             CLASS_S_DESC[dsc$w_length]  = 23;
;  1879          2005  2             CLASS_S_DESC[dsc$a_pointer] = .DESTINATION [DSC$A_POINTER];
;  1880          2006  2             IF NOT (SYS$ASCTIM (TEMP, CLASS_S_DESC, .SOURCE [DSC$A_POINTER], 0))
;  1881          2007  2             THEN
;  1882          2008  2                 SIGNAL( DBG$_DELTIMTOO );
;  1883          2009  2             OUTPUT_STR_LEN = .TEMP;
;  1884          2010  2             END
;  1885          2011  2         ELSE
;  1886          2012  2             IF (.DESTINATION [DSC$B_DTYPE] EQL DSC$K_DTYPE_ADT)  AND              ! Text to ADT
;  1887          2013  2                (.SOURCE [DSC$B_DTYPE] EQL DSC$K_DTYPE_T)
;  1888          2014  2             THEN
;  1889          2015  2                 BEGIN
;  1890          2016  2                 CLASS_S_DESC[dsc$b_class]   = dsc$k_class_s;                     ! Build the string descripto
;  1891          2017  2                 CLASS_S_DESC[dsc$b_dtype]   = dsc$k_dtype_t;
;  1892          2018  2                 CLASS_S_DESC [DSC$W_LENGTH] = .SOURCE [DSC$W_LENGTH];
;  1893          2019  2                 CLASS_S_DESC [DSC$A_POINTER] = .SOURCE [DSC$A_POINTER];
;  1894          2020  2                 IF NOT (SYS$BINTIM( CLASS_S_DESC, .DESTINATION [DSC$A_POINTER]))
;  1895          2021  2                 THEN
;  1896          2022  2                     SIGNAL( DBG$_ABSDATSYN );                                    ! He didn't like to text for
;  1897          2023  2                 END
;  1898          2024  2             ELSE
;  1899          2025  2                 SIGNAL( DBG$_ILLTYPE )                                           ! Nothing but ADT to text or
;  1900          2026  2     ELSE
;  1901          2027  2     BEGIN
;  1902          2028  2     ! Get the conversion rounding flag.  TRUE = round.
;  1903          2029  2     !
;  1904          2030  2     IF ACTUALCOUNT() GTR 3
;  1905          2031  2     THEN
;  1906          2032  2         CVT_ROUND_FLAG = ACTUALPARAMETER(4)
;  1907          2033  2     ELSE
;  1908          2034  2         CVT_ROUND_FLAG = FALSE;
;  1909          2035  2
;  1910          2036  2
;  1911          2037  2     ! Strip the non-significant zeros for packed decimal.
;  1912          2038  2     !
;  1913          2039  2     IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_P AND
;  1914          2040  2        .DESTINATION[DSC$B_DTYPE] NEQ DSC$K_DTYPE_T
;  1915          2041  2     THEN
;  1916          2042  2         SOURCE = DBG$STRIP_ZEROES(.SOURCE);
;  1917          2043  2
;  1918          2044  2
;  1919          2045  2     ! This flag is so that a special error can
;  1920          2046  2     ! be signalled for reserved operand during a decimal string conversion.
;  1921          2047  3     ! Note that this test relies on the fact that the dtypes for the
```

```
;  1922          2048  1     ! decimal string data types are cover the range from the MIN_DEC_DTYPE
;  1923          2049  1     ! code to the MAX_DEC_DTYPE code.
;  1924          2050  1     !
;  1925          2051  1     DECIMAL_CONVERT =
;  1926          2052  1        (.SOURCE[DSC$B_DTYPE] GEQ MIN_DEC_DTYPE) AND
;  1927          2053  1        (.SOURCE[DSC$B_DTYPE] LEQ MAX_DEC_DTYPE);
;  1928          2054  1
;  1929          2055  1
;  1930          2056  1     ! DESTINATION_PTR is used to indicate the destination
;  1931          2057  1     ! of the converted data.  If the data type is unaligned, then
;  1932          2058  1     ! the output-buffer pointer points to a temporary buffer.  Else,
;  1933          2059  1     ! the output-buffer pointer points to the caller's buffer.
;  1934          2060  1     !
;  1935          2061  1     IF .DESTINATION[DSC$B_CLASS] EQL DSC$K_CLASS_UBS
;  1936          2062  1     THEN
;  1937          2063  1         BEGIN
;  1938          2064  1         OUTPUT = OUTPUT_BUFFER;
;  1939          2065  1         DESTINATION_PTR = .DESTINATION[DSC$A_POINTER];
;  1940          2066  1         END
;  1941          2067  1     ELSE
;  1942          2068  1         OUTPUT = .DESTINATION[DSC$A_POINTER];
;  1943          2069  1
;  1944          2070  1
;  1945          2071  1     ! Zero and blank out these records for FIND_CVT_PATH.
;  1946          2072  1     !
;  1947          2073  1     CH$FILL (0, K_SRC_INFO_LENGTH, SRC_INFO);
;  1948          2074  1     CH$FILL (0, K_DST_INFO_LENGTH, DST_INFO);
;  1949          2075  1     CH$FILL (0, K_INTMED_DATA_LENGTH, INTMED_DATA);
;  1950          2076  1     CH$FILL (%C' ', K_TEMP_BUF_LENGTH, TEMP_BUF1);
;  1951          2077  1     CH$FILL (%C' ', K_TEMP_BUF_LENGTH, TEMP_BUF2);
;  1952          2078  1     OUTPUT_STR_LEN = 0;
;  1953          2079  1
;  1954          2080  1
;  1955          2081  1     ! This descriptor is always class S, dtype T.
;  1956          2082  1     ! It is used on various occasions to call routines that require
;  1957          2083  1     ! descriptors as their parameters.
;  1958          2084  1     !
;  1959          2085  1     CLASS_S_DESC [DSC$B_DTYPE] = DSC$K_DTYPE_T;
;  1960          2086  1     CLASS_S_DESC [DSC$B_CLASS] = DSC$K_CLASS_S;
;  1961          2087  1
;  1962          2088  1
;  1963          2089  1     ! Initialize some constants.
;  1964          2090  1     !
;  1965          2091  1     LRGST_P_LU = UPLIT (%P'+4294967295');
;  1966          2092  1     LRGST_D_LU = UPLIT (%D'+4294967295');
;  1967          2093  1     LRGST_H_LU = UPLIT (%H'+4294967295');
;  1968          2094  1     PACK_ZERO = UPLIT (%P'+0');
;  1969          2095  1
;  1970          2096  1
;  1971          2097  1     ! SRC_INFO structure will contain the information about the source data.  In
;  1972          2098  1     ! most cases it will point to the INTMED_DATA buffer because the source data is
;  1973          2099  1     ! usually converted to an intermediate, so before calling FIND_CVT_PATH we
;  1974          2100  1     ! set up the pointer and length fields of SRC_INFO to be INTMED_DATA.
;  1975          2101  1     !
;  1976          2102  1     SRC_INFO [S_POINTER] = INTMED_DATA;
;  1977          2103  1     SRC_INFO [S_LEN] = K_INTMED_DATA_LENGTH;
;  1978          2104  1     }
```

```
;  1979          2105  3
;  1980          2106  3        ! Call FIND_CVT_PATH to get information on the source and destination
;  1981          2107  3        ! (SRC_INFO and DST_INFO), and to determine the conversion path
;  1982          2108  3        ! (CVT_PATH).
;  1983          2109  3        !
;  1984          2110  3        STATUS = FIND_CVT_PATH (.SOURCE, .DESTINATION, SRC_INFO, DST_INFO, CVT_PATH);
;  1985          2111  3
;  1986          2112  3
;  1987          2113  3        ! If we got an error returned to us by FIND_CVT_PATH, it means that one of the
;  1988          2114  3        ! descriptors - SOURCE or DESTINATION - was invalid to this routine.
;  1989          2115  3        ! Errors are represented as negative values.  They are listed in the completion
;  1990          2116  3        ! status section of FIND_CVT_PATH.  Although we get a variety of errors,
;  1991          2117  3        ! from -1 to -7, overlapping can occur.
;  1992          2118  3        !
;  1993          2119  3        IF .STATUS LSS 0
;  1994          2120  3        THEN
;  1995          2121  4            BEGIN
;  1996          2122  4            CASE .STATUS FROM K_INVNBDS TO K_UNSCLAROU OF
;  1997          2123  4                SET
;  1998          2124  4                [K_UNSDTYSTA, K_UNSDTYROU]:  $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  invalid dtype in descriptor')
;  1999          2125  4                [K_UNSCLASTA, K_UNSCLAROU]:  $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  invalid class in descriptor')
;  2000          2126  4                [K_UNSDESSTA, K_UNSDESROU]:  $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  invalid class-dtype combinati
;  2001          2127  4                [K_INVNBDS]:                 $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  invalid numeric byte string d
;  2002          2128  4                TES;
;  2003          2129  4            END;
;  2004          2130  3
;  2005          2131  3
;  2006          2132  3        ! Enable all arithmetic traps, and figure out the scale fator to be used by
;  2007          2133  3        ! the main CASE statement below. The scale factor in SCALE will be a decimal
;  2008          2134  3        ! scale factor. The scale factor in BIN_SCALE will be a binary scale factor.
;  2009          2135  3        !
;  2010          2136  3        BISPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  2011          2137  3        SCALE = (IF .SRC_INFO[S_BIN_SCALE] THEN 0 ELSE .SRC_INFO [S_SCALE]) -
;  2012          2138  3                (IF .DST_INFO[D_BIN_SCALE] THEN 0 ELSE .DST_INFO [D_SCALE]);
;  2013          2139  3        BIN_SCALE = (IF .SRC_INFO[S_BIN_SCALE] THEN .SRC_INFO [S_SCALE] ELSE 0) -
;  2014          2140  3                    (IF .DST_INFO[D_BIN_SCALE] THEN .DST_INFO [D_SCALE] ELSE 0);
;  2015          2141  3
;  2016          2142  3
;  2017          2143  3        ! We now have SRC_INFO, DST_INFO, and CVT_PATH, and the source data
;  2018          2144  3        ! has been converted to an intermediate type.  Next step:  to go from
;  2019          2145  3        ! the intermediate form to a scaled version to the actual data type
;  2020          2146  3        ! called for by the destinaton descriptor.
;  2021          2147  3        !
;  2022          2148  3        ! The following explains the objective of the conversions:
;  2023          2149  3        !
;  2024          2150  3        !           The objective is to convert from intermediate data type provided by
;  2025          2151  3        !           FIND_CVT_PATH routine to the data type that the user has requested in
;  2026          2152  3        !           the destination descriptor.
;  2027          2153  3        !
;  2028          2154  3        !           The intermediate data is in INTMED_DATA, except for when source is
;  2029          2155  3        !           of data type T.  FIND_CVT_PATH does not convert or transform the T
;  2030          2156  3        !           data types, so the intermediate data for this data type is described
;  2031          2157  3        !           by the SOURCE descriptor itself.
;  2032          2158  3        !
;  2033          2159  3        !           The first step is to scale the intermediate data.  The scale is
;  2034          2160  3        !           calculated as:    SCALE = (source scale) - (destination scale).
;  2035          2161  3        !           Scaling cannot always be done on the intermediate data because there
```

```
;  2036          2162  !           may be under/over flow, so scaling is done on either the intermediate
;  2037          2163  !           or the higest data type of the category that the destination data type
;  2038          2164  !           falls in.  The data type with greater range is always selected.
;  2039          2165  !           Caution is taken not to select a scaling intermediate
;  2040          2166  !           data type that requires G, H, or O instructions, unless source or
;  2041          2167  !           destination is of these types.
;  2042          2168  !           At the beginning of each sub-case statement, there is a macro;
;  2043          2169  !           each macro is type specific, and scales the intermediate data type
;  2044          2170  !           involved in that sub-case.
;  2045          2171  !           Regardless of whether there is scaling involved or not the intermediate
;  2046          2172  !           data type is converted to scaling intermediate data type.
;  2047          2173  !           The scaled intermediate data will again end up in INTMED_DATA buffer.
;  2048          2174  !
;  2049          2175  !           Macros that do this scaling are called M_SCALE_x_y:  convert x to y, where
;  2050          2176  !           the result value in y is scaled according to the scale specified
;  2051          2177  !           in source and destination descriptors.
;  2052          2178  !
;  2053          2179  !           The next step is to convert the scaled intermediate data to destination
;  2054          2180  !           data type and move it to where the destination address points to.
;  2055          2181  !           This is done as close to a 'interrupt proof' manner as possible.
;  2056          2182  !           Since only NBDS can be of semantics other than fixed, only in case of
;  2057          2183  !           NBDS (or just text) is the destination copied via a RTL call (LIB$SCOPY_x).
;  2058          2184  !
;  2059          2185  !           PSW is masked such that IV, FU, DV bits are set.
;  2060          2186  !
;  2061          2187      CASE .CVT_PATH FROM K_SMLINT_SMLINT TO K_NBDS_NBDS OF
;  2062          2188          SET
```

```
;  2064          2189  3
;  2065          2190  3          [K_SMLINT_SMLINT]:
;  2066          2191  4              BEGIN
;  2067          2192  4              M_SCALE_L_L;
;  2068          2193  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  2069          2194  4                  SET
;  2070          2195  4
;  2071          2196  4                  [DSC$K_DTYPE_BU]:
;  2072          2197  5                      BEGIN
;  2073          2198  5                      IF (OUTPUT [BYTE_1] = .INTMED_DATA [LONG_1]) GTRU K_LRGST_BU
;  2074          2199  5                          THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2075          2200  4                      END;
;  2076          2201  4
;  2077          2202  4                  [DSC$K_DTYPE_WU]:
;  2078          2203  5                      BEGIN
;  2079          2204  5                      IF (OUTPUT [WORD_1] = .INTMED_DATA [S_LONG_1]) GTRU K_LRGST_WU
;  2080          2205  5                          THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2081          2206  4                      END;
;  2082          2207  4
;  2083          2208  4                  [DSC$K_DTYPE_B]:
;  2084          2209  4                      CVTLB (INTMED_DATA, .OUTPUT);
;  2085          2210  4
;  2086          2211  4                  [DSC$K_DTYPE_W]:
;  2087          2212  4                      CVTLW (INTMED_DATA, .OUTPUT);
;  2088          2213  4
;  2089          2214  4                  [DSC$K_DTYPE_L]:
;  2090          2215  4                      OUTPUT [LONG_1] = .INTMED_DATA [S_LONG_1];
;  2091          2216  4
;  2092          2217  4                  [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  2093          2218  5                      BEGIN
;  2094          2219  5                      MAP
;  2095          2220  5                          OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  2096          2221  5                          INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  2097          2222  5
;  2098          2223  5                      INCR I FROM 0 TO .DST_INFO[D_LEN]-1 DO
;  2099          2224  6                          BEGIN
;  2100          2225  6                          OUTPUT[.I] = .INTMED_DATA[.I];
;  2101          2226  5                          END;
;  2102          2227  4                      END;
;  2103          2228  4
;  2104          2229  4                  [INRANGE, OUTRANGE]:
;  2105          2230  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_smlint');
;  2106          2231  4                  TES;                                !For SMLINT_SMLINT
;  2107          2232  3          END;
```

```
;  2109          2233  3
;  2110          2234  3          [K_SMLINT_LRGINT, K_LRGINT_LRGINT]:
;  2111          2235  4              BEGIN
;  2112          2236  4              SELECTONE .CVT_PATH OF
;  2113          2237  4                  SET
;  2114          2238  4
;  2115          2239  4                  [K_SMLINT_LRGINT]:
;  2116          2240  5                      BEGIN
;  2117          2241  5                      M_SCALE_L_OU;
;  2118          2242  4                      END;
;  2119          2243  4
;  2120          2244  4                  [K_LRGINT_LRGINT]:
;  2121          2245  5                      BEGIN
;  2122          2246  5                      M_SCALE_OU_OU;
;  2123          2247  4                      END;
;  2124          2248  4                  TES;
;  2125          2249  4
;  2126          2250  4              SELECTONE .DESTINATION [DSC$B_DTYPE] OF
;  2127          2251  4                  SET
;  2128          2252  4
;  2129          2253  4                  [DSC$K_DTYPE_LU]:
;  2130          2254  5                      BEGIN
;  2131          2255  5                      IF (.INTMED_DATA [LONG_2] OR .INTMED_DATA [LONG_3] OR .INTMED_DATA [LONG_4]) NEQ 0
;  2132          2256  5                          THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2133          2257  5                      OUTPUT [LONG_1] = .INTMED_DATA [LONG_1];
;  2134          2258  5                      IF .SRC_INFO[S_SIGN]
;  2135          2259  5                      THEN
;  2136          2260  5                          OUTPUT[LONG_1] = -.OUTPUT[S_LONG_1];
;  2137          2261  4                      END;
;  2138          2262  4
;  2139          2263  4                  [DSC$K_DTYPE_Q, DSC$K_DTYPE_OU]:
;  2140          2264  5                      BEGIN
;  2141          2265  5                      IF (.INTMED_DATA [LONG_3] OR .INTMED_DATA [LONG_4]) NEQ 0
;  2142          2266  5                          THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2143          2267  5                      IF .SRC_INFO [S_SIGN]
;  2144          2268  5                      THEN
;  2145          2269  5                          IF .INTMED_DATA [LONG_1] EQL 0
;  2146          2270  5                          THEN
;  2147          2271  6                              BEGIN
;  2148          2272  6                              IF .INTMED_DATA [LONG_2] NEQU %X'80000000'
;  2149          2273  6                              THEN
;  2150          2274  7                                  BEGIN
;  2151          2275  7                                  INTMED_DATA [LONG_2] = .INTMED_DATA [LONG_2] XOR %X'FFFFFFFF';
;  2152          2276  7                                  INTMED_DATA [LONG_2] = .INTMED_DATA [LONG_2] + 1;
;  2153          2277  6                                  END;
;  2154          2278  6                              END
;  2155          2279  5                          ELSE
;  2156          2280  6                              BEGIN
;  2157          2281  6                              INTMED_DATA [LONG_1] = .INTMED_DATA [LONG_1] XOR %X'FFFFFFFF';
;  2158          2282  6                              INTMED_DATA [LONG_2] = .INTMED_DATA [LONG_2] XOR %X'FFFFFFFF';
;  2159          2283  6                              INTMED_DATA [LONG_1] = .INTMED_DATA [LONG_1] + 1;
;  2160          2284  5                              END;
;  2161          2285  5                      OUTPUT [LONG_1] = .INTMED_DATA [LONG_1];
;  2162          2286  5                      OUTPUT [LONG_2] = .INTMED_DATA [LONG_2];
;  2163          2287  4                      END;
;  2164          2288  4
;  2165          2289  4                  [DSC$K_DTYPE_O]:
```

```
;  2166          2290  5                BEGIN
;  2167          2291  5
;  2168          2292  5                ! The S_SIGN field is set if we need to negate the octaword.
;  2169          2293  5                !
;  2170          2294  5                IF .SRC_INFO[S_SIGN]
;  2171          2295  5                THEN
;  2172          2296  6                    BEGIN
;  2173          2297  6
;  2174          2298  6                    ! Check for %X'80000000000000000000000000000000'
;  2175          2299  6                    ! This should not go through the code below, but rather
;  2176          2300  6                    ! just be deposited.
;  2177          2301  6                    !
;  2178          2302  7                    IF NOT (.INTMED_DATA[LONG_1] EQL 0 AND
;  2179          2303  7                            .INTMED_DATA[LONG_2] EQL 0 AND
;  2180          2304  7                            .INTMED_DATA[LONG_3] EQL 0 AND
;  2181          2305  7                            .INTMED_DATA[LONG_4] EQL %X'80000000')
;  2182          2306  6                    THEN
;  2183          2307  7                        BEGIN
;  2184          2308  7                        MAP
;  2185          2309  7                            INTMED_DATA:  VECTOR[K_INTMED_DATA_LENGTH/4];
;  2186          2310  7
;  2187          2311  7                        ! Take the one's complement.
;  2188          2312  7                        !
;  2189          2313  7                        INCR NEXT_LONGWORD FROM 0 TO 3 DO
;  2190          2314  7                            INTMED_DATA[.NEXT_LONGWORD] =
;  2191          2315  7                                .INTMED_DATA[.NEXT_LONGWORD] XOR %X'FFFFFFFF';
;  2192          2316  7
;  2193          2317  7                        ! Add 1 to the result.
;  2194          2318  7                        !
;  2195          2319  7                        INCR NEXT_LONGWORD FROM 0 TO 3 DO
;  2196          2320  7                            IF .INTMED_DATA[.NEXT_LONGWORD] EQLU %X'FFFFFFFF'
;  2197          2321  7                            THEN
;  2198          2322  7                                INTMED_DATA[.NEXT_LONGWORD] = 0
;  2199          2323  7                            ELSE
;  2200          2324  8                                BEGIN
;  2201          2325  8                                INTMED_DATA[.NEXT_LONGWORD] =
;  2202          2326  8                                    .INTMED_DATA[.NEXT_LONGWORD] + 1;
;  2203          2327  8                                EXITLOOP;
;  2204          2328  7                                END;
;  2205          2329  6                        END;
;  2206          2330  5                    END;
;  2207          2331  5
;  2208          2332  5                CH$MOVE (16, INTMED_DATA, .OUTPUT);
;  2209          2333  4                END;
;  2210          2334  4
;  2211          2335  4            [OTHERWISE]:
;  2212          2336  4                SELECTONE .CVT_PATH OF
;  2213          2337  4                SET
;  2214          2338  4                [K_SMLINT_LRGINT]:
;  2215          2339  4                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_lrgint');
;  2216          2340  4
;  2217          2341  4                [K_LRGINT_LRGINT]:
;  2218          2342  4                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint_lrgint');
;  2219          2343  4                TES;
;  2220          2344  4            TES;                                !For SMLINT_LRGINT, LRGINT_LRGINT.
;  2221          2345  3        END;
```

```
;  2223          2346  3
;  2224          2347  3        [K_SMLINT_SMLFLTCMPLX, K_LRGINT_SMLFLTCMPLX, K_SMLFLTCMPLX_SMLFLTCMPLX, K_DEC_SMLFLTCMPLX, K_NBDS_SM
;  2225          2348  4            BEGIN
;  2226          2349  4            SELECTONE .CVT_PATH OF
;  2227          2350  4                SET
;  2228          2351  4
;  2229          2352  4                [K_SMLINT_SMLFLTCMPLX]:
;  2230          2353  5                    BEGIN
;  2231          2354  5                    M_SCALE_L_D;
;  2232          2355  4                    END;
;  2233          2356  4
;  2234          2357  4                [K_LRGINT_SMLFLTCMPLX]:
;  2235          2358  5                    BEGIN
;  2236          2359  5                    M_SCALE_OU_D;
;  2237          2360  4                    END;
;  2238          2361  4
;  2239          2362  4                [K_SMLFLTCMPLX_SMLFLTCMPLX]:
;  2240          2363  5                    BEGIN
;  2241          2364  5                    M_SCALE_D_D;
;  2242          2365  4                    END;
;  2243          2366  4
;  2244          2367  4                [K_DEC_SMLFLTCMPLX]:
;  2245          2368  5                    BEGIN
;  2246          2369  5                    M_SCALE_P_D;
;  2247          2370  4                    END;
;  2248          2371  4
;  2249          2372  4                [K_NBDS_SMLFLTCMPLX]:
;  2250          2373  5                    BEGIN
;  2251          2374  5                    CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  2252          2375  5                    CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  2253          2376  5                    STATUS = OTS$CVT_T_D (CLASS_S_DESC, INTMED_DATA, 0, -.SCALE,
;  2254          2377  5                        (K_IGN_BLKS_OR_K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  2255          2378  5                    IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  2256          2379  4                    END;
;  2257          2380  4                TES;
;  2258          2381  4
;  2259          2382  4            CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_F TO DSC$K_DTYPE_D OF
;  2260          2383  4                SET
;  2261          2384  4
;  2262          2385  4                [DSC$K_DTYPE_F]:
;  2263          2386  4                    CVTDF (INTMED_DATA, .OUTPUT);
;  2264          2387  4
;  2265          2388  4                [DSC$K_DTYPE_D]:
;  2266          2389  5                    BEGIN
;  2267          2390  5                    OUTPUT [LONG_1] = .INTMED_DATA [LONG_1];
;  2268          2391  5                    OUTPUT [LONG_2] = .INTMED_DATA [LONG_2];
;  2269          2392  4                    END;
;  2270          2393  4
;  2271          2394  4                [INRANGE, OUTRANGE]:
;  2272          2395  4                    CASE .DESTINATION[DSC$B_DTYPE] FROM DSC$K_DTYPE_FC TO DSC$K_DTYPE_DC OF
;  2273          2396  4                        SET
;  2274          2397  4
;  2275          2398  4                        [DSC$K_DTYPE_FC]:
;  2276          2399  5                            BEGIN
;  2277          2400  5                            CVTDF (INTMED_DATA, .OUTPUT);
;  2278          2401  5                            CVTDF (INTMED_DATA+8, .OUTPUT+4);
;  2279          2402  4                            END;
```

```
;  2280          2403  4                    [DSC$K_DTYPE_DC]:
;  2281          2404  4                        CH$MOVE (16, INTMED_DATA, .OUTPUT);
;  2282          2405  4
;  2283          2406  4                    [INRANGE, OUTRANGE]:
;  2284          2407  4                        SELECTONE .CVT_PATH OF
;  2285          2408  4                            SET
;  2286          2409  4                            [K_SMLINT_SMLFLTCMPLX]:
;  2287          2410  4                                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_smlfltcmplx');
;  2288          2411  4                            [K_LRGINT_SMLFLTCMPLX]:
;  2289          2412  4                                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint_smlfltcmplx');
;  2290          2413  4                            [K_SMLFLTCMPLX_SMLFLTCMPLX]:
;  2291          2414  4                                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx_smlfltcmplx');
;  2292          2415  4                            [K_DEC_SMLFLTCMPLX]:
;  2293          2416  4                                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_smlfltcmplx');
;  2294          2417  4                            [K_NBDS_SMLFLTCMPLX]:
;  2295          2418  4                                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_smlfltcmplx');
;  2296          2419  4                            TES;
;  2297          2420  4                        TES;
;  2298          2421  4                TES;                                !For SMLINT_SMLFLTCMPLX, LRGINT_SMLFLTCMPLX, SMLFLTCMPLX_SML
;  2299          2422  4
;  2300          2423  4            IF .SRC_INFO [S_SIGN] THEN OUTPUT [0, 15, 1, 0] = 1;
;  2301          2424  4            END;
;  2302          2425  3
```

```
;  2304          2426  3
;  2305          2427  3          [K_SMLINT_LRGFLTCMPLX, K_LRGINT_LRGFLTCMPLX, K_SMLFLTCMPLX_LRGFLTCMPLX, K_LRGFLTCMPLX_LRGFLTCMPLX, K
;  2306          2428  4              BEGIN
;  2307          2429  4              SELECTONE .CVT_PATH OF
;  2308          2430  4                  SET
;  2309          2431  4
;  2310          2432  4                  [K_SMLINT_LRGFLTCMPLX]:
;  2311          2433  5                      BEGIN
;  2312          2434  5                      M_SCALE_L_H;
;  2313          2435  4                      END;
;  2314          2436  4
;  2315          2437  4                  [K_LRGINT_LRGFLTCMPLX]:
;  2316          2438  5                      BEGIN
;  2317          2439  5                      M_SCALE_OU_H;
;  2318          2440  4                      END;
;  2319          2441  4
;  2320          2442  4                  [K_SMLFLTCMPLX_LRGFLTCMPLX]:
;  2321          2443  5                      BEGIN
;  2322          2444  5                      M_SCALE_D_H;
;  2323          2445  4                      END;
;  2324          2446  4
;  2325          2447  4                  [K_LRGFLTCMPLX_LRGFLTCMPLX]:
;  2326          2448  5                      BEGIN
;  2327          2449  5                      IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_G OR
;  2328          2450  5                         .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  2329          2451  5                      THEN
;  2330          2452  6                          M_SCALE_G_H
;  2331          2453  5                      ELSE
;  2332          2454  5                          M_SCALE_H_H;
;  2333          2455  4                      END;
;  2334          2456  4
;  2335          2457  4                  [K_DEC_LRGFLTCMPLX]:
;  2336          2458  5                      BEGIN
;  2337          2459  5                      M_SCALE_P_H;
;  2338          2460  4                      END;
;  2339          2461  4                  TES;
;  2340          2462  4
;  2341          2463  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_G TO DSC$K_DTYPE_H OF
;  2342          2464  4                  SET
;  2343          2465  4
;  2344          2466  4                  [DSC$K_DTYPE_G]:
;  2345          2467  4                      CVTHG (INTMED_DATA, .OUTPUT);
;  2346          2468  4
;  2347          2469  4                  [DSC$K_DTYPE_H]:
;  2348          2470  4                      CH$MOVE (16, INTMED_DATA, .OUTPUT);
;  2349          2471  4
;  2350          2472  4                  [INRANGE, OUTRANGE]:
;  2351          2473  4                      CASE .DESTINATION[DSC$B_DTYPE] FROM DSC$K_DTYPE_GC TO DSC$K_DTYPE_HC OF
;  2352          2474  4                          SET
;  2353          2475  4
;  2354          2476  4                          [DSC$K_DTYPE_GC]:
;  2355          2477  5                              BEGIN
;  2356          2478  5                              CVTHG (INTMED_DATA, .OUTPUT);
;  2357          2479  5                              CVTHG (INTMED_DATA+16, .OUTPUT+8);
;  2358          2480  4                              END;
;  2359          2481  4
;  2360          2482  4                          [DSC$K_DTYPE_HC]:
```

```
;  2361          2483  4                    CH$MOVE (32, INTMED_DATA, .OUTPUT);
;  2362          2484  4
;  2363          2485  4                [INRANGE, OUTRANGE]:
;  2364          2486  4                    SELECTONE .CVT_PATH OF
;  2365          2487  4                        SET
;  2366          2488  4                        [K_SMLINT_LRGFLTCMPLX]:
;  2367          2489  4                            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_lrgfltcmplx');
;  2368          2490  4                        [K_LRGINT_LRGFLTCMPLX]:
;  2369          2491  4                            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint_lrgfltcmplx');
;  2370          2492  4                        [K_SMLFLTCMPLX_LRGFLTCMPLX]:
;  2371          2493  4                            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx_lrgfltcmplx');
;  2372          2494  4                        [K_LRGFLTCMPLX_LRGFLTCMPLX]:
;  2373          2495  4                            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx_lrgfltcmplx');
;  2374          2496  4                        [K_DEC_LRGFLTCMPLX]:
;  2375          2497  4                            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_lrgfltcmplx');
;  2376          2498  4                        TES;
;  2377          2499  4                    TES;
;  2378          2500  4            TES;                !For SMLINT_LRGFLTCMPLX, LRGINT_LRGFLTCMPLX, SMLFLTCMPLX_LRGFLTCMPLX, LRGFLT
;  2379          2501  4
;  2380          2502  4        IF .SRC_INFO [S_SIGN] THEN OUTPUT [0, 15, 1, 0] = 1;
;  2381          2503  3        END;
```

```
;  2383          2504  3
;  2384          2505  3          [K_SMLINT_DEC, K_DEC_DEC]:
;  2385          2506  4              BEGIN
;  2386          2507  4              SELECTONE .CVT_PATH OF
;  2387          2508  4                  SET
;  2388          2509  4
;  2389          2510  4                  [K_SMLINT_DEC]:
;  2390          2511  5                      BEGIN
;  2391          2512  5                      M_SCALE_L_P;
;  2392          2513  4                      END;
;  2393          2514  4
;  2394          2515  4                  [K_DEC_DEC]:
;  2395          2516  5                      BEGIN
;  2396          2517  5                      M_SCALE_P_P;
;  2397          2518  4                      END;
;  2398          2519  4                  TES;
;  2399          2520  4
;  2400          2521  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_NU TO DSC$K_DTYPE_P OF
;  2401          2522  4                  SET
;  2402          2523  4
;  2403          2524  4                  [DSC$K_DTYPE_NU]:
;  2404          2525  5                      BEGIN
;  2405          2526  5                      IF .SRC_INFO [S_SIGN] THEN SIGNAL (DBG$_CVTNEGUNS, 1, .DBG$GL_OPCODE_NAME);
;  2406          2527  5                      CVTPT (NO_DIGITS, INTMED_DATA, LIB$AB_CVTPT_U, DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2407          2528  4                      END;
;  2408          2529  4
;  2409          2530  4                  [DSC$K_DTYPE_NL]:
;  2410          2531  4                      CVTPS (NO_DIGITS, INTMED_DATA,
;  2411          2532  4                          %REF (
;  2412          2533  4                              IF .DESTINATION [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .DESTINATION [DSC$W_LENGTH] - 1
;  2413          2534  4                          ), .OUTPUT);
;  2414          2535  4
;  2415          2536  4                  [DSC$K_DTYPE_NLO]:
;  2416          2537  5                      BEGIN
;  2417          2538  5                      CVTPT (NO_DIGITS, INTMED_DATA, LIB$AB_CVTPT_U, DESTINATION [DSC$W_LENGTH], TEMP_BUF1);
;  2418          2539  7                      TEMP_BUF1 [BYTE_1] = (IF .SRC_INFO [S_SIGN] THEN .(.TEMP_BUF1 [BYTE_1] + LIB$AB_CVT_U_O
;  2419          2540  5                          48 + 10) ELSE .(.TEMP_BUF1 [BYTE_1] + LIB$AB_CVT_U_O - 48));
;  2420          2541  5                      CH$MOVE (.DESTINATION [DSC$W_LENGTH], TEMP_BUF1, .OUTPUT);
;  2421          2542  4                      END;
;  2422          2543  4
;  2423          2544  4                  [DSC$K_DTYPE_NR]:
;  2424          2545  5                      BEGIN
;  2425          2546  5                      LOCAL
;  2426          2547  5                          DES_LEN;
;  2427          2548  5                      DES_LEN =
;  2428          2549  6                      BEGIN
;  2429          2550  6                      IF .DESTINATION [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .DESTINATION [DSC$W_LENGTH] - 1
;  2430          2551  5                      END;
;  2431          2552  5                      CVTPS (NO_DIGITS, INTMED_DATA, DES_LEN, TEMP_BUF1);
;  2432          2553  5                      BLOCK [INTMED_DATA + .DES_LEN, 0, 0, 8, 0;, BYTE] = .TEMP_BUF1 [BYTE_1];
;  2433          2554  5                      CH$MOVE (.DES_LEN, TEMP_BUF1 + 1, INTMED_DATA);
;  2434          2555  5                      CH$MOVE (.DES_LEN + 1, INTMED_DATA, .OUTPUT);
;  2435          2556  4                      END;
;  2436          2557  4
;  2437          2558  4                  [DSC$K_DTYPE_NRO, DSC$K_DTYPE_NZ]:
;  2438          2559  4                      CVTPT (NO_DIGITS, INTMED_DATA,
;  2439          2560  5                          (IF .DESTINATION [DSC$B_DTYPE] EQL DSC$K_DTYPE_NRO THEN LIB$AB_CVTPT_O ELSE
```

```
;  2440          2561  4                          LIB$AB_CVTPT_Z), DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2441          2562  4
;  2442          2563  4                  [DSC$K_DTYPE_P]:
;  2443          2564  5                      BEGIN
;  2444          2565  5 !                    CVTPS (NO_DIGITS, INTMED_DATA, DESTINATION [DSC$W_LENGTH], TEMP_BUF1);
;  2445          2566  5 !                    CVTSP (DESTINATION [DSC$W_LENGTH], TEMP_BUF1, DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2446          2567  5                      CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF1);
;  2447          2568  5                      CVTSP (NO_DIGITS, TEMP_BUF1, DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2448          2569  4                      END;
;  2449          2570  4
;  2450          2571  4                  [INRANGE, OUTRANGE]:
;  2451          2572  4                      SELECTONE .CVT_PATH OF
;  2452          2573  4                          SET
;  2453          2574  4                          [K_SMLINT_DEC]:
;  2454          2575  4                              $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_dec');
;  2455          2576  4                          [K_DEC_DEC]:
;  2456          2577  4                              $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_dec');
;  2457          2578  4                          TES;
;  2458          2579  4                      TES;                                !For SMLINT_DEC, DEC_DEC
;  2459          2580  3              END;
```

```
;  2461          2581  3          [K_LRGINT_SMLINT]:
;  2462          2582  3              BEGIN
;  2463          2583  4              M_SCALE_OU_OU;
;  2464          2584  4              IF (.INTMED_DATA [LONG_2] OR .INTMED_DATA [LONG_3] OR .INTMED_DATA [LONG_4]) NEQ 0
;  2465          2585  4              THEN
;  2466          2586  4                  SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2467          2587  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  2468          2588  4                  SET
;  2469          2589  4
;  2470          2590  4                  [DSC$K_DTYPE_BU]:
;  2471          2591  4                      BEGIN
;  2472          2592  5                      IF .INTMED_DATA [BYTE_2] OR .INTMED_DATA [WORD_2] NEQ 0
;  2473          2593  5                          THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2474          2594  5                      OUTPUT [BYTE_1] = .INTMED_DATA [LONG_1];
;  2475          2595  5                      END;
;  2476          2596  4
;  2477          2597  4                  [DSC$K_DTYPE_WU]:
;  2478          2598  4                      BEGIN
;  2479          2599  5                      IF .INTMED_DATA [WORD_2] NEQ 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2480          2600  5                      OUTPUT [WORD_1] = .INTMED_DATA [LONG_1];
;  2481          2601  5                      END;
;  2482          2602  4
;  2483          2603  4                  [DSC$K_DTYPE_B]:
;  2484          2604  4                      BEGIN
;  2485          2605  5                      IF .INTMED_DATA [S_LONG_1] LSS 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2486          2606  5                      IF .SRC_INFO [S_SIGN] THEN INTMED_DATA [LONG_1] = -.INTMED_DATA [S_LONG_1];
;  2487          2607  5                      CVTLB (INTMED_DATA, .OUTPUT);
;  2488          2608  5                      END;
;  2489          2609  4
;  2490          2610  4                  [DSC$K_DTYPE_W]:
;  2491          2611  4                      BEGIN
;  2492          2612  5                      IF .INTMED_DATA [S_LONG_1] LSS 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2493          2613  5                      IF .SRC_INFO [S_SIGN] THEN INTMED_DATA [LONG_1] = -.INTMED_DATA [S_LONG_1];
;  2494          2614  5                      CVTLW (INTMED_DATA, .OUTPUT);
;  2495          2615  5                      END;
;  2496          2616  4
;  2497          2617  4                  [DSC$K_DTYPE_L]:
;  2498          2618  4                      BEGIN
;  2499          2619  5                      IF .INTMED_DATA [S_LONG_1] EQL K_LRGST_NEG_L AND .SRC_INFO [S_SIGN] EQL 1
;  2500          2620  5                      THEN
;  2501          2621  5                          OUTPUT [LONG_1] = .INTMED_DATA [S_LONG_1]
;  2502          2622  5                      ELSE
;  2503          2623  5                          BEGIN
;  2504          2624  6                          IF .INTMED_DATA [S_LONG_1] LSS 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  2505          2625  6                          IF .SRC_INFO [S_SIGN] THEN INTMED_DATA [LONG_1] = -.INTMED_DATA [S_LONG_1];
;  2506          2626  6                          OUTPUT [LONG_1] = .INTMED_DATA [S_LONG_1];
;  2507          2627  6                          END;
;  2508          2628  5                      END;
;  2509          2629  4
;  2510          2630  4                  [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  2511          2631  4                      BEGIN
;  2512          2632  5                      MAP
;  2513          2633  5                          OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  2514          2634  5                          INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  2515          2635  5
;  2516          2636  5                      INCR I FROM 0 TO .DST_INFO[D_LEN] - 1 DO
;  2517          2637  5
```

```
;  2518          2638  6                    BEGIN
;  2519          2639  6                    OUTPUT[.I] = .INTMED_DATA[.I];
;  2520          2640  5                    END;
;  2521          2641  4                END;
;  2522          2642  4
;  2523          2643  4            [INRANGE, OUTRANGE]:
;  2524          2644  4                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint smlint');
;  2525          2645  4            TES;                                !For LRGINT_SMLINT
;  2526          2646  3        END;
```

```
;  2528          2647  3
;  2529          2648  3          [K_LRGINT_DEC, K_SMLFLTCMPLX_DEC, K_LRGFLTCMPLX_DEC, K_NBDS_DEC]:
;  2530          2649  4              BEGIN
;  2531          2650  4              CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  2532          2651  4              CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  2533          2652  4              SELECTONE .CVT_PATH OF
;  2534          2653  4                  SET
;  2535          2654  4
;  2536          2655  4                  [K_LRGINT_DEC]:
;  2537          2656  5                      BEGIN
;  2538          2657  5                      CVTROUM (INTMED_DATA, TEMP_BUF1);
;  2539          2658  5                      IF .SRC_INFO [S_SIGN] THEN TEMP_BUF1<15, 1, 0> = 1;
;  2540          2659  5                      STATUS = FOR$CVT_H_TF (TEMP_BUFT, CLASS_S_DESC, 0, .SCALE, 0, 0, 1);
;  2541          2660  4                      END;
;  2542          2661  4
;  2543          2662  4                  [K_SMLFLTCMPLX_DEC]:
;  2544          2663  5                      BEGIN
;  2545          2664  5                      IF .INTMED_DATA<15, 1, 0> THEN SRC_INFO [S_SIGN] = 1;
;  2546          2665  5                      STATUS = FOR$CVT_D_TF (INTMED_DATA, CLASS_S_DESC, 0, .SCALE, 0, 0, 1);
;  2547          2666  4                      END;
;  2548          2667  4
;  2549          2668  4                  [K_LRGFLTCMPLX_DEC]:
;  2550          2669  5                      BEGIN
;  2551          2670  5                      IF .INTMED_DATA<15, 1, 0> THEN SRC_INFO [S_SIGN] = 1;
;  2552          2671  5                      IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_G OR
;  2553          2672  5                          .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  2554          2673  5                      THEN
;  2555          2674  5                          STATUS = FOR$CVT_G_TF (INTMED_DATA, CLASS_S_DESC, 0, .SCALE, 0, 0, 1)
;  2556          2675  5                      ELSE
;  2557          2676  5                          STATUS = FOR$CVT_H_TF (INTMED_DATA, CLASS_S_DESC, 0, .SCALE, 0, 0, 1);
;  2558          2677  4                      END;
;  2559          2678  4
;  2560          2679  4                  [K_NBDS_DEC]:
;  2561          2680  5                      BEGIN
;  2562          2681  5                      CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  2563          2682  5                      CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  2564          2683  5                      STATUS = OTS$CVT_T_R (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  2565          2684  5                          (K_IGN_BLKS OR K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  2566          2685  5                      IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  2567          2686  5                      IF .TEMP_BUF1<15, 1, 0> THEN SRC_INFO [S_SIGN] = 1;
;  2568          2687  5                      CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  2569          2688  5                      CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  2570          2689  5                      STATUS = FOR$CVT_H_TF (TEMP_BUF1, CLASS_S_DESC, 0, 0, 0, 0, 1);
;  2571          2690  4                      END;
;  2572          2691  4              TES;
;  2573          2692  4
;  2574          2693  4          IF NOT .STATUS THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCODE_NAME);
;  2575          2694  4          BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  2576          2695  4          NO_DIGITS = K_TEMP_BUF_LENGTH - .BUF_OFFSET - 2;
;  2577          2696  4
;  2578          2697  4          CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_NU TO DSC$K_DTYPE_P OF
;  2579          2698  4              SET
;  2580          2699  4
;  2581          2700  4              [DSC$K_DTYPE_NU]:
;  2582          2701  5                  BEGIN
;  2583          2702  5                  IF .SRC_INFO [S_SIGN] THEN SIGNAL (DBG$_CVTNEGUNS, 1, .DBG$GL_OPCODE_NAME);
;  2584          2703  5                  IF .NO_DIGITS GTR .DESTINATION [DSC$W_LENGTH] THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCOD
```

```
;  2585          2704  5                CH$FILL (%X'30', .DESTINATION [DSC$W_LENGTH] - .NO_DIGITS, TEMP_BUF1);
;  2586          2705  5                CH$MOVE (.NO_DIGITS, TEMP_BUF2 + .BUF_OFFSET + 1,
;  2587          2706  5                    TEMP_BUF1 + .DESTINATION [DSC$W_LENGTH] - .NO_DIGITS);
;  2588          2707  5                CH$MOVE (.DESTINATION [DSC$W_LENGTH], TEMP_BUF1, .OUTPUT);
;  2589          2708  4                END;
;  2590          2709  4
;  2591          2710  4            [DSC$K_DTYPE_NL]:
;  2592          2711  5                BEGIN
;  2593          2712  5                LOCAL
;  2594          2713  5                    DES_LEN;
;  2595          2714  5                DES_LEN =
;  2596          2715  6                BEGIN
;  2597          2716  6                IF .DESTINATION [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .DESTINATION [DSC$W_LENGTH] - 1
;  2598          2717  5                END;
;  2599          2718  5                IF .DES_LEN LSS .NO_DIGITS THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCODE_NAME);
;  2600          2719  5                CVTSP (NO_DIGITS, TEMP_BUF2 + .BUF_OFFSET, DES_LEN, TEMP_BUF1);
;  2601          2720  5                CVTPS (DES_LEN, TEMP_BUF1, DES_LEN, .OUTPUT);
;  2602          2721  4                END;
;  2603          2722  4
;  2604          2723  4            [DSC$K_DTYPE_NLO]:
;  2605          2724  5                BEGIN
;  2606          2725  5                CH$FILL (%X'30', .BUF_OFFSET + 1, TEMP_BUF2);
;  2607          2726  5                IF .NO_DIGITS GTR .DESTINATION [DSC$W_LENGTH] THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCOD
;  2608          2727  5                BUF_OFFSET = K_TEMP_BUF_LENGTH - .DESTINATION [DSC$W_LENGTH] - 1;
;  2609          2728  7                BLOCK [TEMP_BUF2 + .BUF_OFFSET, 0, 0, 8, 0;, BYTE] = (IF .SRC_INFO [S_SIGN] THEN .(.BLOC
;  2610          2729  7                    [TEMP_BUF2 + .BUF_OFFSET, 0, 0, 8, 0;, BYTE] + LIB$AB_CVT_U_0 - 48 + 10) ELSE .(
;  2611          2730  5                    .BLOCK [TEMP_BUF2 + .BUF_OFFSET, 0, 0, 8, 0;, BYTE] + LIB$AB_CVT_U_0 - 48));
;  2612          2731  5                CH$MOVE (.DESTINATION [DSC$W_LENGTH], TEMP_BUF2 + .BUF_OFFSET, .OUTPUT);
;  2613          2732  4                END;
;  2614          2733  4
;  2615          2734  4            [DSC$K_DTYPE_NR]:
;  2616          2735  5                BEGIN
;  2617          2736  5                LOCAL
;  2618          2737  5                    DES_LEN;
;  2619          2738  5                DES_LEN =
;  2620          2739  6                BEGIN
;  2621          2740  6                IF .DESTINATION [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .DESTINATION [DSC$W_LENGTH] - 1
;  2622          2741  5                END;
;  2623          2742  5                IF .NO_DIGITS GTR .DES_LEN THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCODE_NAME);
;  2624          2743  5                CH$FILL (%X'30', .DES_LEN - .NO_DIGITS + 1, TEMP_BUF1);
;  2625          2744  5                CH$MOVE (.NO_DIGITS, TEMP_BUF2 + .BUF_OFFSET + 1, TEMP_BUF1 + .DES_LEN - .NO_DIGITS);
;  2626          2745  5                BLOCK [TEMP_BUF1 + .DES_LEN, 0, 0, 8, 0;, BYTE] = .BLOCK [TEMP_BUF2 + .BUF_OFFSET, 0,
;  2627          2746  5                    0, 8, 0;, BYTE];
;  2628          2747  5                CH$MOVE (.DES_LEN + 1, TEMP_BUF1, .OUTPUT);
;  2629          2748  4                END;
;  2630          2749  4
;  2631          2750  4            [DSC$K_DTYPE_NRO, DSC$K_DTYPE_NZ]:
;  2632          2751  5                BEGIN
;  2633          2752  5                IF .NO_DIGITS GTR .DESTINATION [DSC$W_LENGTH] THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCOD
;  2634          2753  5                CVTSP (NO_DIGITS, TEMP_BUF2 + .BUF_OFFSET, DESTINATION [DSC$W_LENGTH], TEMP_BUF1);
;  2635          2754  5                CVTPT (DESTINATION [DSC$W_LENGTH], TEMP_BUF1,
;  2636          2755  6                    (IF .DESTINATION [DSC$B_DTYPE] EQL DSC$K_DTYPE_NRO THEN LIB$AB_CVTPT_0 ELSE
;  2637          2756  5                        LIB$AB_CVTPT_Z), DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2638          2757  4                END;
;  2639          2758  4
;  2640          2759  4            [DSC$K_DTYPE_P]:
;  2641          2760  5                BEGIN
```

```
;  2642          2761  5                 IF .NO_DIGITS GTR 31 THEN SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCODE_NAME);
;  2643          2762  5                 CVTSP (NO_DIGITS, TEMP_BUF2 + .BUF_OFFSET, DESTINATION [DSC$W_LENGTH], .OUTPUT);
;  2644          2763  4                 END;
;  2645          2764  4
;  2646          2765  4             [INRANGE, OUTRANGE]:
;  2647          2766  4                 SELECTONE .CVT_PATH OF
;  2648          2767  4                     SET
;  2649          2768  4                     [K_LRGINT_DEC]:
;  2650          2769  4                         $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint_dec');
;  2651          2770  4                     [K_SMLFLTCMPLX_DEC]:
;  2652          2771  4                         $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx_dec');
;  2653          2772  4                     [K_LRGFLTCMPLX_DEC]:
;  2654          2773  4                         $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx_dec');
;  2655          2774  4                     [K_NBDS_DEC]:
;  2656          2775  4                         $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_dec');
;  2657          2776  4                     TES;
;  2658          2777  4                 TES;                                !For LRGINT_DEC, SMLFLTCMPLX_DEC, LRGFLTCMPLX_DEC, NBDS_DEC.
;  2659          2778  3             END;
```

```
;  2661          2779  3                [K_SMLINT_NBDS, K_LRGINT_NBDS, K_DEC_NBDS]:
;  2662          2780  3                    SELECTONE .DESTINATION [DSC$B_DTYPE] OF
;  2663          2781  3                        SET
;  2664          2782  3
;  2665          2783  3                    [DSC$K_DTYPE_BU, DSC$K_DTYPE_T, DSC$K_DTYPE_VT, DSC$K_DTYPE_AC, DSC$K_DTYPE_AZ]:
;  2666          2784  3                        BEGIN
;  2667          2785  4                        CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  2668          2786  4                        CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  2669          2787  4
;  2670          2788  4                        ! Compute 'DIGITS_IN_FRACT' based on scale.
;  2671          2789  4                        ! For negative scales, the number of digits in the fraction
;  2672          2790  4                        ! is just the absolute value of the scale. This seems
;  2673          2791  4                        ! to work for both binary and decimal scales. For example,
;  2674          2792  4                        ! (binary 101 with scale factor -2) = binary 1.01 =
;  2675          2793  4                        ! 1 + 0/2 + 1/4 = 1.25, which has 2 digits in the fraction.
;  2676          2794  4                        ! For non-negative scale, DIGITS_IN_FRACT is zero.
;  2677          2795  4                        ! First do a consistency check to ensure we do not have
;  2678          2796  4                        ! both decimal and binary scale factors - if we do,
;  2679          2797  4                        ! something is wrong.
;  2680          2798  4                        !
;  2681          2799  4                        DIGITS_IN_FRACT = 0;
;  2682          2800  4                        IF (.BIN_SCALE NEQ 0) AND (.SCALE NEQ 0)
;  2683          2801  5                        THEN
;  2684          2802  4                            $DBG_ERROR('DBGCVTDX\DBG$CVT_DX_DX inconsistent scale factors');
;  2685          2803  4                        IF .BIN_SCALE LSS 0
;  2686          2804  4                        THEN
;  2687          2805  4                            DIGITS_IN_FRACT = -.BIN_SCALE;
;  2688          2806  4                        IF .SCALE LSS 0
;  2689          2807  4                        THEN
;  2690          2808  4                            DIGITS_IN_FRACT = -.SCALE;
;  2691          2809  4
;  2692          2810  4                        SELECTONE .CVT_PATH OF
;  2693          2811  4                            SET
;  2694          2812  4
;  2695          2813  4                            [K_SMLINT_NBDS]:
;  2696          2814  4                                BEGIN
;  2697          2815  5                                CVTLD (INTMED_DATA, TEMP_BUF1);
;  2698          2816  5                                !
;  2699          2817  5                                ! Take care of binary scale factors by doing
;  2700          2818  5                                ! the divide or multiply.
;  2701          2819  5                                !
;  2702          2820  5                                WHILE .BIN_SCALE LSS 0 DO
;  2703          2821  5                                    BEGIN
;  2704          2822  6                                    DIVD2(UPLIT (%D'2.0'), TEMP_BUF1);
;  2705          2823  6                                    BIN_SCALE = .BIN_SCALE + 1;
;  2706          2824  6                                    END;
;  2707          2825  5                                WHILE .BIN_SCALE GTR 0 DO
;  2708          2826  5                                    BEGIN
;  2709          2827  6                                    MULD2(UPLIT (%D'2.0'), TEMP_BUF1);
;  2710          2828  6                                    BIN_SCALE = .BIN_SCALE - 1;
;  2711          2829  6                                    END;
;  2712          2830  5
;  2713          2831  5                                STATUS = FOR$CVT_D_TF (TEMP_BUF1, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE);
;  2714          2832  5                                END;
;  2715          2833  4
;  2716          2834  4                            [K_LRGINT_NBDS]:
;  2717          2835  4
```

```
;  2718          2836  4                IF .SOURCE[ DSC$B_DTYPE ] NEQ DSC$K_DTYPE_O                    ! A004
;  2719          2837  4                THEN                                                           ! A004
;  2720          2838  4                    BEGIN
;  2721          2839  4                    CVTROUH (INTMED_DATA, TEMP_BUF1);
;  2722          2840  4                    IF .SRC_INFO [S_SIGN] THEN TEMP_BUF1<15, 1, 0> = 1;
;  2723          2841  4
;  2724          2842  4                    ! Take care of binary scale factors by doing
;  2725          2843  4                    ! the divide or multiply.
;  2726          2844  4                    !
;  2727          2845  4                    WHILE .BIN_SCALE LSS 0 DO
;  2728          2846  4                        BEGIN
;  2729          2847  4                        DIVH2(UPLIT (%H'2.0'), TEMP_BUF1);
;  2730          2848  4                        BIN_SCALE = .BIN_SCALE + 1;
;  2731          2849  4                        END;
;  2732          2850  4                    WHILE .BIN_SCALE GTR 0 DO
;  2733          2851  4                        BEGIN
;  2734          2852  4                        MULH2(UPLIT (%H'2.0'), TEMP_BUF1);
;  2735          2853  4                        BIN_SCALE = .BIN_SCALE - 1;
;  2736          2854  4                        END;
;  2737          2855  4
;  2738          2856  4                    STATUS = FOR$CVT_H_TF (TEMP_BUF1, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE);
;  2739          2857  4                    END
;  2740          2858  4                ELSE                                                           ! A004
;  2741          2859  4                    BEGIN                                                      ! A004
;  2742          2860  4                    LOCAL                                                      ! A004
;  2743          2861  4                        Previous_Value : VECTOR[4];                            ! A004
;  2744          2862  4
;  2745          2863  4                    MAP                                                        ! A004
;  2746          2864  4                        INTMED_DATA : VECTOR[4];                               ! A004
;  2747          2865  4
;  2748          2866  4                    ! Don't support scale factor on octaword.
;  2749          2867  4                    !
;  2750          2868  4                    IF .BIN_SCALE NEQ 0
;  2751          2869  4                    THEN
;  2752          2870  4                        $DBG_ERROR('DBGCVTDX\DBG$CVT_DX_DX scale factor on octaword not supporte
;  2753          2871  4
;  2754          2872  4                    CLASS_S_DESC[ DSC$W_LENGTH ] = 0;                          ! A004
;  2755          2873  4
;  2756          2874  4                    !++
;  2757          2875  4                    ! Init the Previous value
;  2758          2876  4                    !--
;  2759          2877  4                    CH$MOVE( 16,                                               ! A004
;  2760          2878  4                             CH$PTR( INTMED_DATA),                             ! A004
;  2761          2879  4                             CH$PTR( Previous_value ) );                       ! A004
;  2762          2880  4
;  2763          2881  4                    !++
;  2764          2882  4                    ! By dividing the value by ten and multiplying it by
;  2765          2883  4                    !    ten the original value and the new value may be
;  2766          2884  4                    !    subtracted to obtain the value of the least
;  2767          2885  4                    !    significant digit.
;  2768          2886  4                    ! Repeating allows the building up of the string
;  2769          2887  4                    !    from the back.
;  2770          2888  4                    !--
;  2771          2889  4                    DO                                                         ! A004
;  2772          2890  5                        BEGIN                                                  ! A004
;  2773          2891  6                        LOCAL                                                  ! A004
;  2774          2892  6
```

```
;  2775          2893  6          Saved_Value : VECTOR[4];                                ! A004
;  2776          2894  6
;  2777          2895  6      !++
;  2778          2896  6      ! Save the previous value
;  2779          2897  6      !--
;  2780          2898  6      CH$MOVE( 16,                                                 ! A004
;  2781          2899  6               CH$PTR( INTMED_DATA),                               ! A004
;  2782          2900  6               CH$PTR( Previous_value ) );                         ! A004
;  2783          2901  6
;  2784          2902  6      !++
;  2785          2903  6      ! Divide by ten
;  2786          2904  6      !--
;  2787          2905  6      DBG$CVT_SCALE_OU_DOWN_BY_10_R1( INTMED_DATA );              ! A004
;  2788          2906  6
;  2789          2907  6      !++
;  2790          2908  6      ! Save the divided value for the next time
;  2791          2909  6      !--
;  2792          2910  6      CH$MOVE( 16,                                                 ! A004
;  2793          2911  6               CH$PTR( INTMED_DATA),                               ! A004
;  2794          2912  6               CH$PTR( Saved_value ) );                            ! A004
;  2795          2913  6
;  2796          2914  6      !++
;  2797          2915  6      ! Multiply by ten to remove for the subtraction
;  2798          2916  6      !--
;  2799          2917  6      DBG$CVT_SCALE_OU_UP_BY_10_R1( INTMED_DATA );                ! A004
;  2800          2918  6
;  2801          2919  6      !++
;  2802          2920  6      ! Move the previous digits down
;  2803          2921  6      !--
;  2804          2922  6      DECR Current_position FROM .CLASS_S_DESC[DSC$W_LENGTH] - 1
;  2805          2923  6                           TO 0 DO
;  2806          2924  6          CH$WCHAR( CH$RCHAR( CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER ] + .Curren
;  2807          2925  6                    CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER ] + .Current_position
;  2808          2926  6
;  2809          2927  6      !++
;  2810          2928  6      ! Subtract and put the new digit in the string
;  2811          2929  6      !--
;  2812          2930  6      CH$WCHAR(.Previous_value[0]-.INTMED_DATA[0] + %C'0',        ! A004
;  2813          2931  6               CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER] ) );         ! A004
;  2814          2932  6
;  2815          2933  6      !++
;  2816          2934  6      ! Increment the length
;  2817          2935  6      !--
;  2818          2936  6      CLASS_S_DESC[ DSC$W_LENGTH ] =                               ! A004
;  2819          2937  6              .CLASS_S_DESC[ DSC$W_LENGTH ] + 1;                   ! A004
;  2820          2938  6
;  2821          2939  6      !++
;  2822          2940  6      ! Saved value becomes the previous value
;  2823          2941  6      !--
;  2824          2942  6      CH$MOVE( 16,                                                 ! A004
;  2825          2943  6               CH$PTR( Saved_value),                               ! A004
;  2826          2944  6               CH$PTR( INTMED_DATA ) );                            ! A004
;  2827          2945  6
;  2828          2946  6      END                                                          ! A004
;  2829          2947  5  WHILE (.INTMED_DATA[ 3 ] NEQ 0)  OR                             ! A004
;  2830          2948  5        (.INTMED_DATA[ 2 ] NEQ 0)  OR                             ! A004
;  2831          2949  5        (.INTMED_DATA[ 1 ] NEQ 0)  OR                             ! A004
```

```
;  2832          2950  5                    (.INTMED_DATA[ 0 ] NEQ 0);                                  ! A004
;  2833          2951  5                                                                                ! A004
;  2834          2952  5                ;++
;  2835          2953  5                ; Load in a '-' if there is one
;  2836          2954  5                ;--
;  2837          2955  5                IF .SRC_INFO[ S_SIGN ]
;  2838          2956  5                THEN
;  2839          2957  6                    BEGIN
;  2840          2958  6                    DECR Current_position FROM .CLASS_S_DESC[DSC$W_LENGTH] - 1
;  2841          2959  6                                          TO 0 DO
;  2842          2960  6                        CH$WCHAR( CH$RCHAR( CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER ] + .Curren
;  2843          2961  6                                  CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER ] + .Current_position
;  2844          2962  6                    CH$WCHAR( %C'-',
;  2845          2963  6                              CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER] ) );
;  2846          2964  6                    CLASS_S_DESC[ DSC$W_LENGTH ] = .CLASS_S_DESC[ DSC$W_LENGTH ] + 1;
;  2847          2965  5                    END;
;  2848          2966  5
;  2849          2967  5                ;++
;  2850          2968  5                ; Put a '.' on the end just like CVTROUH
;  2851          2969  5                ;--
;  2852          2970  5                CH$WCHAR(%C'.', CH$PTR( .CLASS_S_DESC[ DSC$A_POINTER] + ! A004
;  2853          2971  5                                        .CLASS_S_DESC[ DSC$W_LENGTH ]));! A004
;  2854          2972  5
;  2855          2973  5                STATUS = SS$_NORMAL;                                            ! A004
;  2856          2974  4                END;                                                            ! A004
;  2857          2975  4
;  2858          2976  4            [K_DEC_NBDS]:
;  2859          2977  5                BEGIN
;  2860          2978  5
;  2861          2979  5                ! Don't support binary scale factor on packed.
;  2862          2980  5                !
;  2863          2981  5                IF .BIN_SCALE NEQ 0
;  2864          2982  5                THEN
;  2865          2983  5                    $DBG_ERROR('DBGCVTDX\DBG$CVT_DX_DX binary scale factor on packed not support
;  2866          2984  5
;  2867          2985  5                NO_DIGITS = .SRC_INFO [S_LEN];
;  2868          2986  5                CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF2);
;  2869          2987  5                CLASS_S_DESC [DSC$W_LENGTH] = .NO_DIGITS + 1;
;  2870          2988  5                OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF1, 0, 0,
;  2871          2989  5                    (K_IGN_BLKS OR K_ENB_UNDERFLOW OR K_IGN_TABS ));
;  2872          2990  5                STATUS = FOR$CVT_H_TF (TEMP_BUF1, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE);
;  2873          2991  4                END;
;  2874          2992  4            TES;
;  2875          2993  4
;  2876          2994  4        BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  2877          2995  4        NEXT_BLANK = CH$FIND_CH (K_TEMP_BUF_LENGTH-.BUF_OFFSET, TEMP_BUF2+.BUF_OFFSET, %C' ');
;  2878          2996  4        IF .NEXT_BLANK EQL 0
;  2879          2997  4        THEN
;  2880          2998  4            FINAL_LEN = K_TEMP_BUF_LENGTH - .BUF_OFFSET
;  2881          2999  4        ELSE
;  2882          3000  4            FINAL_LEN = .NEXT_BLANK - .BUF_OFFSET - TEMP_BUF2;
;  2883          3001  4        IF .DIGITS_IN_FRACT EQL 0
;  2884          3002  4        THEN
;  2885          3003  4            FINAL_LEN = .FINAL_LEN - 1;
;  2886          3004  4
;  2887          3005  4        IF NOT .STATUS
;  2888          3006  4        THEN
```

```
;  2889     3007  5                    BEGIN
;  2890     3008  5                    CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  2891     3009  5                    IF .CVT_PATH EQL K_DEC_NBDS
;  2892     3010  5                    THEN
;  2893     3011  5                        DIGITS_IN_FRACT = 31
;  2894     3012  5                    ELSE
;  2895     3013  5                        IF .DST_INFO [D_LEN] - 9 LEQ 0
;  2896     3014  5                        THEN
;  2897     3015  5                            DIGITS_IN_FRACT = 33
;  2898     3016  5                        ELSE
;  2899     3017  5                            DIGITS_IN_FRACT = MIN (33, .DST_INFO [D_LEN] - 9);
;  2900     3018  5                    STATUS = FOR$CVT_A_TE (TEMP_BUF1, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE, 0, 4);
;  2901     3019  5                    IF NOT .STATUS THEN $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  error in h-to-te conversio
;  2902     3020  5                    BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  2903     3021  5                    FINAL_LEN = K_TEMP_BUF_LENGTH - .BUF_OFFSET;
;  2904     3022  4                    END;
;  2905     3023  4
;  2906     3024  4                OUTPUT_STR_LEN = .FINAL_LEN;
;  2907     3025  4                SELECTONE .DESTINATION[DSC$B_DTYPE] OF
;  2908     3026  4                    SET
;  2909     3027  4                    [DSC$K_DTYPE_AC]:
;  2910     3028  5                        BEGIN
;  2911     3029  5                        MAP
;  2912     3030  5                            OUTPUT:  REF VECTOR[, BYTE];
;  2913     3031  5                        CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  2914     3032  5                        CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[1];
;  2915     3033  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  2916     3034  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  2917     3035  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  2918     3036  5                        OUTPUT[0] = .FINAL_LEN;
;  2919     3037  4                        END;
;  2920     3038  4
;  2921     3039  4                    [DSC$K_DTYPE_AZ]:
;  2922     3040  5                        BEGIN
;  2923     3041  5                        MAP
;  2924     3042  5                            OUTPUT:  REF VECTOR[, BYTE];
;  2925     3043  5                        CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  2926     3044  5                        CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[0];
;  2927     3045  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  2928     3046  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  2929     3047  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  2930     3048  5                        OUTPUT[.FINAL_LEN + 1] = 0;
;  2931     3049  4                        END;
;  2932     3050  4
;  2933     3051  4                    [OTHERWISE]:
;  2934     3052  5                        BEGIN
;  2935     3053  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, .DESTINATION);
;  2936     3054  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  2937     3055  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  2938     3056  4                        END;
;  2939     3057  4                    TES;
;  2940     3058  3                END;
;  2941     3059  3
;  2942     3060  3                [OTHERWISE]:
;  2943     3061  3                    SELECTONE .CVT_PATH OF
;  2944     3062  3                        SET
;  2945     3063  3                        [K_SMLINT_NBDS]:
```

```
;  2946          3064  3                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlint_nbds');
;  2947          3065  3                [K_LRGINT_NBDS]:
;  2948          3066  3                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgint_nbds');
;  2949          3067  3                [K_DEC_NBDS]:
;  2950          3068  3                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_nbds');
;  2951          3069  3                TES;
;  2952          3070  3            TES;                            !For SMLINT_NBDS, LRGINT_NBDS, DEC_NBDS
;  2953          3071  3
```

```
;  2955          3072  3
;  2956          3073  3          [K_SMLFLTCMPLX_SMLINT]:
;  2957          3074  4              BEGIN
;  2958          3075  4              M_SCALE_D_D;
;  2959          3076  4              IF .CVT_ROUND_FLAG
;  2960          3077  4              THEN
;  2961          3078  4                  CVTRDL (INTMED_DATA, TEMP_BUF1)
;  2962          3079  4              ELSE
;  2963          3080  4                  CVTDL  (INTMED_DATA, TEMP_BUF1);
;  2964          3081  4
;  2965          3082  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  2966          3083  4                  SET
;  2967          3084  4
;  2968          3085  4                  [DSC$K_DTYPE_BU]:
;  2969          3086  5                      BEGIN
;  2970          3087  5                      IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_BU THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME
;  2971          3088  5                      OUTPUT [BYTE_1] = .TEMP_BUF1 [BYTE_1];
;  2972          3089  4                      END;
;  2973          3090  4
;  2974          3091  4                  [DSC$K_DTYPE_WU]:
;  2975          3092  5                      BEGIN
;  2976          3093  5                      IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_WU THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME
;  2977          3094  5                      OUTPUT [WORD_1] = .TEMP_BUF1 [WORD_1];
;  2978          3095  4                      END;
;  2979          3096  4
;  2980          3097  4                  [DSC$K_DTYPE_B]:
;  2981          3098  5                      BEGIN
;  2982          3099  5                      CVTLB (TEMP_BUF1, .OUTPUT);
;  2983          3100  4                      END;
;  2984          3101  4
;  2985          3102  4                  [DSC$K_DTYPE_W]:
;  2986          3103  5                      BEGIN
;  2987          3104  5                      CVTLW (TEMP_BUF1, .OUTPUT);
;  2988          3105  4                      END;
;  2989          3106  4
;  2990          3107  4                  [DSC$K_DTYPE_L]:
;  2991          3108  4                      OUTPUT [LONG_1] = .TEMP_BUF1 [S_LONG_1];
;  2992          3109  4
;  2993          3110  4                  [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  2994          3111  5                      BEGIN
;  2995          3112  5                      MAP
;  2996          3113  5                          OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  2997          3114  5                          INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  2998          3115  5
;  2999          3116  5                      INCR I FROM 0 TO .DST_INFO[D_LEN] - 1 DO
;  3000          3117  6                          BEGIN
;  3001          3118  6                          OUTPUT[.I] = .INTMED_DATA[.I];
;  3002          3119  5                          END;
;  3003          3120  4                      END;
;  3004          3121  4
;  3005          3122  4                  [INRANGE, OUTRANGE]:
;  3006          3123  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx_smlint');
;  3007          3124  4                  TES;                                !For SMLFLTCMPLX_SMLINT
;  3008          3125  3          END;
```

```
;  3010          3126  3
;  3011          3127  3          [K_SMLFLTCMPLX_LRGINT]:
;  3012          3128  4              BEGIN
;  3013          3129  4              M_SCALE_D_D;
;  3014          3130  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_LU TO DSC$K_DTYPE_O OF
;  3015          3131  4                  SET
;  3016          3132  4
;  3017          3133  4                  [DSC$K_DTYPE_LU]:
;  3018          3134  5                      BEGIN
;  3019          3135  5                      IF CMPD (INTMED_DATA, .LRGST_D_LU) GTR 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NA
;  3020          3136  5                      BICPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3021          3137  5                      IF .CVT_ROUND_FLAG
;  3022          3138  5                      THEN
;  3023          3139  5                          CVTRDL (INTMED_DATA, .OUTPUT)
;  3024          3140  5                      ELSE
;  3025          3141  5                          CVTDL  (INTMED_DATA, .OUTPUT);
;  3026          3142  5
;  3027          3143  5                      BISPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3028          3144  4                      END;
;  3029          3145  4
;  3030          3146  4                  [DSC$K_DTYPE_Q, DSC$K_DTYPE_QU]:
;  3031          3147  4                      CVTRDQ (INTMED_DATA, .OUTPUT);
;  3032          3148  4
;  3033          3149  4                  [DSC$K_DTYPE_O]:
;  3034          3150  5                      BEGIN
;  3035          3151  5                      CVTDH (INTMED_DATA, TEMP_BUF1);
;  3036          3152  5                      CVTRHO (TEMP_BUF1, .OUTPUT);
;  3037          3153  4                      END;
;  3038          3154  4
;  3039          3155  4                  [INRANGE, OUTRANGE]:
;  3040          3156  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx lrgint');
;  3041          3157  4                  TES;                                !For SMLFLTCMPLX_LRGINT
;  3042          3158  3              END;
```

```
;  3044          3159  3
;  3045          3160  3          [K_SMLFLTCMPLX_NBDS]:
;  3046          3161  3              SELECTONE .DESTINATION [DSC$B_DTYPE] OF
;  3047          3162  3                  SET
;  3048          3163  3
;  3049          3164  3                  [DSC$K_DTYPE_BU, DSC$K_DTYPE_T, DSC$K_DTYPE_VT, DSC$K_DTYPE_AC, DSC$K_DTYPE_AZ]:
;  3050          3165  4                      BEGIN
;  3051          3166  4                      CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  3052          3167  4                      CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  3053          3168  4                      DIGITS_IN_FRACT =
;  3054          3169  5                      BEGIN
;  3055          3170  5                      CASE .SOURCE [DSC$B_DTYPE] FROM DSC$K_DTYPE_F TO DSC$K_DTYPE_D OF
;  3056          3171  5                          SET
;  3057          3172  5
;  3058          3173  5                          [DSC$K_DTYPE_F]:
;  3059          3174  5                              7;
;  3060          3175  5
;  3061          3176  5                          [DSC$K_DTYPE_D]:
;  3062          3177  5                              16;
;  3063          3178  5                          TES
;  3064          3179  4                      END;
;  3065          3180  4                      IF .DST_INFO [D_LEN] - 7 GTR 0
;  3066          3181  4                      THEN
;  3067          3182  4                          DIGITS_IN_FRACT = MIN (.DIGITS_IN_FRACT,
;  3068          3183  4                              .DST_INFO [D_LEN] - 7);
;  3069          3184  4                      STATUS = FOR$CVT_D_TE (INTMED_DATA, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE, 0);
;  3070          3185  4                      IF NOT .STATUS THEN $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  error in d-to-te conversion');
;  3071          3186  4                      BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  3072          3187  4                      FINAL_LEN = K_TEMP_BUF_LENGTH - .BUF_OFFSET;
;  3073          3188  4                      OUTPUT_STR_LEN = .FINAL_LEN;
;  3074          3189  4
;  3075          3190  4                      SELECTONE .DESTINATION[DSC$B_DTYPE] OF
;  3076          3191  4                          SET
;  3077          3192  4                          [DSC$K_DTYPE_AC]:
;  3078          3193  5                              BEGIN
;  3079          3194  5                              MAP
;  3080          3195  5                                  OUTPUT:  REF VECTOR[, BYTE];
;  3081          3196  5                              CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  3082          3197  5                              CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[1];
;  3083          3198  5                              STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  3084          3199  5                              IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, T..DBG$GL_OPCODE_NAME);
;  3085          3200  5                              IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3086          3201  5                              OUTPUT[0] = .FINAL_LEN;
;  3087          3202  4                              END;
;  3088          3203  4
;  3089          3204  4                          [DSC$K_DTYPE_AZ]:
;  3090          3205  5                              BEGIN
;  3091          3206  5                              MAP
;  3092          3207  5                                  OUTPUT:  REF VECTOR[, BYTE];
;  3093          3208  5                              CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  3094          3209  5                              CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[0];
;  3095          3210  5                              STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  3096          3211  5                              IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, T, .DBG$GL_OPCODE_NAME);
;  3097          3212  5                              IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3098          3213  5                              OUTPUT[.FINAL_LEN + 1] = 0;
;  3099          3214  4                              END;
;  3100          3215  4
```

```
;  3101          3216  4                    [OTHERWISE]:
;  3102          3217  5                        BEGIN
;  3103          3218  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, .DESTINATION);
;  3104          3219  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  3105          3220  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3106          3221  4                        END;
;  3107          3222  4                    TES;
;  3108          3223  3                END;
;  3109          3224
;  3110          3225  3            [OTHERWISE]:
;  3111          3226  3                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  smlfltcmplx_nbds');
;  3112          3227  3            TES;                                !For SMLFLTCMPLX_NBDS
```

```
;  3114          3228  3          [K_LRGFLTCMPLX_SMLINT]:
;  3115          3229  3              BEGIN
;  3116          3230  4              IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_G OR
;  3117          3231  4                 .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  3118          3232  4              THEN
;  3119          3233  4                  M_SCALE_G_H
;  3120          3234  5              ELSE
;  3121          3235  4                  M_SCALE_H_H;
;  3122          3236  4              IF .CVT_ROUND_FLAG
;  3123          3237  4              THEN
;  3124          3238  4                  CVTRHL (INTMED_DATA, TEMP_BUF1)
;  3125          3239  4              ELSE
;  3126          3240  4                  CVTHL  (INTMED_DATA, TEMP_BUF1);
;  3127          3241  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  3128          3242  4                  SET
;  3129          3243  4
;  3130          3244  4                  [DSC$K_DTYPE_BU]:
;  3131          3245  4                      BEGIN
;  3132          3246  5                      IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_BU THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME
;  3133          3247  5                      OUTPUT [BYTE_1] = .TEMP_BUF1 [BYTE_1];
;  3134          3248  5                      END;
;  3135          3249  4
;  3136          3250  4                  [DSC$K_DTYPE_WU]:
;  3137          3251  4                      BEGIN
;  3138          3252  5                      IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_WU THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME
;  3139          3253  5                      OUTPUT [WORD_1] = .TEMP_BUF1 [WORD_1];
;  3140          3254  5                      END;
;  3141          3255  4
;  3142          3256  4                  [DSC$K_DTYPE_B]:
;  3143          3257  4                      BEGIN
;  3144          3258  5                      CVTLB (TEMP_BUF1, .OUTPUT);
;  3145          3259  5                      END;
;  3146          3260  4
;  3147          3261  4                  [DSC$K_DTYPE_W]:
;  3148          3262  4                      BEGIN
;  3149          3263  5                      CVTLW (TEMP_BUF1, .OUTPUT);
;  3150          3264  5                      END;
;  3151          3265  4
;  3152          3266  4                  [DSC$K_DTYPE_L]:
;  3153          3267  4                      OUTPUT [[ONG_1] = .TEMP_BUF1 [S_LONG_1];
;  3154          3268  4
;  3155          3269  4                  [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  3156          3270  4                      BEGIN
;  3157          3271  5                      MAP
;  3158          3272  5                          OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  3159          3273  5                          INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  3160          3274  5
;  3161          3275  5                      INCR I FROM 0 TO .DST_INFO[D_LEN] - 1 DO
;  3162          3276  5                          BEGIN
;  3163          3277  6                          OUTPUT[.I] = .INTMED_DATA[.I];
;  3164          3278  6                          END;
;  3165          3279  5                      END;
;  3166          3280  4
;  3167          3281  4                  [INRANGE, OUTRANGE]:
;  3168          3282  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx_smlint');
;  3169          3283  4              TES;                                !For LRGFLTCMPLX_SMLINT
;  3170          3284  4
```

```
; 3171          3285  3          END;
```

```
;  3173          3286  3
;  3174          3287  3          [K_LRGFLTCMPLX_LRGINT]:
;  3175          3288  4              BEGIN
;  3176          3289  4              IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_G OR
;  3177          3290  4                 .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  3178          3291  4              THEN
;  3179          3292  5                  M_SCALE_G_H
;  3180          3293  4              ELSE
;  3181          3294  4                  M_SCALE_H_H;
;  3182          3295  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_LU TO DSC$K_DTYPE_O OF
;  3183          3296  4                  SET
;  3184          3297  4
;  3185          3298  4                  [DSC$K_DTYPE_LU]:
;  3186          3299  5                      BEGIN
;  3187          3300  5                      IF CMPH (INTMED_DATA, .LRGST_H_LU) GTR 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NA
;  3188          3301  5                      BICPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3189          3302  5                      IF .CVT_ROUND_FLAG
;  3190          3303  5                      THEN
;  3191          3304  5                          CVTRHL (INTMED_DATA, .OUTPUT)
;  3192          3305  5                      ELSE
;  3193          3306  5                          CVTHL  (INTMED_DATA, .OUTPUT);
;  3194          3307  5                      BISPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3195          3308  4                      END;
;  3196          3309  4
;  3197          3310  4                  [DSC$K_DTYPE_Q, DSC$K_DTYPE_OU]:
;  3198          3311  4                      CVTRHQ (INTMED_DATA, .OUTPUT);
;  3199          3312  4
;  3200          3313  4                  [DSC$K_DTYPE_O]:
;  3201          3314  4                      CVTRHO (INTMED_DATA, .OUTPUT);
;  3202          3315  4
;  3203          3316  4                  [INRANGE, OUTRANGE]:
;  3204          3317  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx lrgint');
;  3205          3318  4                  TES;                                !For LRGFLTCMPLX_LRGINT
;  3206          3319  3              END;
```

```
;  3208          3320  3
;  3209          3321  3          [K_LRGFLTCMPLX_SMLFLTCMPLX]:
;  3210          3322  4              BEGIN
;  3211          3323  4              IF .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_G OR
;  3212          3324  4                 .SOURCE[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  3213          3325  4              THEN
;  3214          3326  5                  M_SCALE_G_H
;  3215          3327  4              ELSE
;  3216          3328  4                  M_SCALE_H_H;
;  3217          3329  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_F TO DSC$K_DTYPE_D OF
;  3218          3330  4                  SET
;  3219          3331  4
;  3220          3332  4                  [DSC$K_DTYPE_F]:
;  3221          3333  4                      CVTHF (INTMED_DATA, .OUTPUT);
;  3222          3334  4
;  3223          3335  4                  [DSC$K_DTYPE_D]:
;  3224          3336  4                      CVTHD (INTMED_DATA, .OUTPUT);
;  3225          3337  4
;  3226          3338  4                  [INRANGE, OUTRANGE]:
;  3227          3339  4                      CASE .DESTINATION[DSC$B_DTYPE] FROM DSC$K_DTYPE_FC TO DSC$K_DTYPE_DC OF
;  3228          3340  4                          SET
;  3229          3341  4
;  3230          3342  4                          [DSC$K_DTYPE_FC]:
;  3231          3343  5                              BEGIN
;  3232          3344  5                              CVTHF (INTMED_DATA, .OUTPUT);
;  3233          3345  5                              CVTHF (INTMED_DATA+16, .OUTPUT+4);
;  3234          3346  4                              END;
;  3235          3347  4
;  3236          3348  4                          [DSC$K_DTYPE_DC]:
;  3237          3349  5                              BEGIN
;  3238          3350  5                              CVTHD (INTMED_DATA, .OUTPUT);
;  3239          3351  5                              CVTHD (INTMED_DATA+16, .OUTPUT+8);
;  3240          3352  4                              END;
;  3241          3353  4
;  3242          3354  4                          [INRANGE, OUTRANGE]:
;  3243          3355  4                              $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx_smlfltcmplx');
;  3244          3356  4                          TES;
;  3245          3357  4                  TES;                                !For LRGFLTCMPLX_SMLFLTCMPLX
;  3246          3358  3          END;
```

```
;  3248          3359  3          [K_LRGFLTCMPLX_NBDS]:
;  3249          3360  3              SELECTONE .DESTINATION [DSC$B_DTYPE] OF
;  3250          3361  3                  SET
;  3251          3362  3
;  3252          3363  3                  [DSC$K_DTYPE_BU, DSC$K_DTYPE_T, DSC$K_DTYPE_VT, DSC$K_DTYPE_AC, DSC$K_DTYPE_AZ]:
;  3253          3364  3                      BEGIN
;  3254          3365  4                      LOCAL
;  3255          3366  4                          DIGITS_IN_EXP,
;  3256          3367  4                          NOT_DIGITS;
;  3257          3368  4
;  3258          3369  4                      CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  3259          3370  4                      CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  3260          3371  4                      CASE .SOURCE [DSC$B_DTYPE] FROM DSC$K_DTYPE_G TO DSC$K_DTYPE_H OF
;  3261          3372  4                          SET
;  3262          3373  4                          [DSC$K_DTYPE_G]:
;  3263          3374  4                              BEGIN
;  3264          3375  5                              DIGITS_IN_FRACT = 15;
;  3265          3376  5                              DIGITS_IN_EXP = 3;
;  3266          3377  5                              NOT_DIGITS = 7;
;  3267          3378  5                              IF .DST_INFO [D_LEN] - .NOT_DIGITS GTR 0
;  3268          3379  5                              THEN
;  3269          3380  5                                  DIGITS_IN_FRACT = MIN (.DIGITS_IN_FRACT, .DST_INFO [D_LEN] - .NOT_DIGITS);
;  3270          3381  5                              STATUS = FOR$CVT_G_TE (INTMED_DATA, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE, 0, .
;  3271          3382  5                              IF NOT .STATUS THEN $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  error in g-to-te conve
;  3272          3383  5                              END;
;  3273          3384  4
;  3274          3385  4                          [DSC$K_DTYPE_H]:
;  3275          3386  4                              BEGIN
;  3276          3387  5                              DIGITS_IN_FRACT = 33;
;  3277          3388  5                              DIGITS_IN_EXP = 4;
;  3278          3389  5                              NOT_DIGITS = 8;
;  3279          3390  5                              IF .DST_INFO [D_LEN] - .NOT_DIGITS GTR 0
;  3280          3391  5                              THEN
;  3281          3392  5                                  DIGITS_IN_FRACT = MIN (.DIGITS_IN_FRACT, .DST_INFO [D_LEN] - .NOT_DIGITS);
;  3282          3393  5                              STATUS = FOR$CVT_H_TE (INTMED_DATA, CLASS_S_DESC, .DIGITS_IN_FRACT, .SCALE, 0, .
;  3283          3394  5                              IF NOT .STATUS THEN $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  error in h-to-te conve
;  3284          3395  5                              END;
;  3285          3396  4                      TES;
;  3286          3397  4
;  3287          3398  4                  BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  3288          3399  4                  FINAL_LEN = K_TEMP_BUF_LENGTH - .BUF_OFFSET;
;  3289          3400  4                  OUTPUT_STR_LEN = .FINAL_LEN;
;  3290          3401  4
;  3291          3402  4                  SELECTONE .DESTINATION[DSC$B_DTYPE] OF
;  3292          3403  4                      SET
;  3293          3404  4                      [DSC$K_DTYPE_AC]:
;  3294          3405  4                          BEGIN
;  3295          3406  5                          MAP
;  3296          3407  5                              OUTPUT:  REF VECTOR[, BYTE];
;  3297          3408  5                          CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  3298          3409  5                          CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[1];
;  3299          3410  5                          STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  3300          3411  5                          IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  3301          3412  5                          IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3302          3413  5                          OUTPUT[0] = .FINAL_LEN;
;  3303          3414  5                          END;
;  3304          3415  4
```

```
;  3305        3416  4                    [DSC$K_DTYPE_AZ]:
;  3306        3417  4                        BEGIN
;  3307        3418  5                        MAP
;  3308        3419  5                            OUTPUT:  REF VECTOR[, BYTE];
;  3309        3420  5                        CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  3310        3421  5                        CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[0];
;  3311        3422  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_DESC);
;  3312        3423  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  3313        3424  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3314        3425  5                        OUTPUT[.FINAL_LEN + 1] = 0;
;  3315        3426  5                        END;
;  3316        3427  4
;  3317        3428  4                    [OTHERWISE]:
;  3318        3429  4                        BEGIN
;  3319        3430  5                        STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, .DESTINATION);
;  3320        3431  5                        IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME);
;  3321        3432  5                        IF NOT .STATUS THEN SIGNAL (.STATUS);
;  3322        3433  5                        END;
;  3323        3434  4                    TES;
;  3324        3435  4                END;
;  3325        3436  3
;  3326        3437  3            [OTHERWISE]:
;  3327        3438  3                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  lrgfltcmplx_nbds');
;  3328        3439  3            TES;                                !For LRGFLTCMPLX_NBDS
;  3329        3440  3
```

```
;  3331     3441  3
;  3332     3442  3          [K_DEC_SMLINT]:
;  3333     3443  4              BEGIN
;  3334     3444  4
;  3335     3445  4              IF .DESTINATION [DSC$V_FL_BINSCALE]
;  3336     3446  4              THEN
;  3337     3447  5                  BEGIN
;  3338     3448  5                  ! This is a HACK for scaled binary.  The Idea is to run the
;  3339     3449  5                  ! scaled packed decimal up to H_Float and then back down to
;  3340     3450  5                  ! the particular dtype below.  The algorithm is as follows:
;  3341     3451  5                  !
;  3342     3452  5                  ! The destination is a binary scale type so the conversion is
;  3343     3453  5                  ! done by hand.
;  3344     3454  5                  ! 1) Get the sign.
;  3345     3455  5                  ! 2) Get the scale of the H_Float.
;  3346     3456  5                  ! 3) Check if an overflow will occur.  An underflow is
;  3347     3457  5                  !       acceptable and will be truncated automatically.
;  3348     3458  5                  ! 4) Move the most significant H_Float fractional bits
;  3349     3459  5                  !       into the temporary destination.
;  3350     3460  5                  !       (Note: this includes the redundant most significant
;  3351     3461  5                  !       fraction bit.
;  3352     3462  5                  ! 5) Alter the destination to the correct scale.
;  3353     3463  5                  ! 6) This is an absolute value so correct for the sign.
;  3354     3464  5                  ! 7) Move the result into the final destination.
;  3355     3465  5                  !
;  3356     3466  5                  ! ****************** HACK - BAB Dec. 1983 ********************
;  3357     3467  5                  !
;  3358     3468  5                  M_SCALE_P_H;
;  3359     3469  5                  END
;  3360     3470  4              ELSE
;  3361     3471  5                  BEGIN
;  3362     3472  5                  M_SCALE_P_P;
;  3363     3473  5                  CVTPL (NO_DIGITS, INTMED_DATA, TEMP_BUF1);
;  3364     3474  4                  END;
;  3365     3475  4
;  3366     3476  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  3367     3477  4                  SET
;  3368     3478  4
;  3369     3479  4                  [DSC$K_DTYPE_BU]:
;  3370     3480  5                      BEGIN
;  3371     3481  5
;  3372     3482  5                      ! If the target is not a binary scale, then just move the
;  3373     3483  5                      ! converted value in.
;  3374     3484  5                      !
;  3375     3485  5                      IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3376     3486  5                      THEN
;  3377     3487  6                          BEGIN
;  3378     3488  6                          IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_BU
;  3379     3489  6                          THEN
;  3380     3490  6                              SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  3381     3491  6                          OUTPUT [BYTE_1] = .TEMP_BUF1 [BYTE_1];
;  3382     3492  6                          END
;  3383     3493  5                      ELSE
;  3384     3494  5
;  3385     3495  5                          ! If the sign and the scale of the H_Float are zero,
;  3386     3496  5                          ! then the value is zero.
;  3387     3497  5                          !
```

```
;  3388          3498  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3389          3499  5                    THEN
;  3390          3500  5                        OUTPUT[BYTE_1] = 0
;  3391          3501  5                    ELSE
;  3392          3502  6                        BEGIN
;  3393          3503  6                        TEMP_BUF1 = 0;
;  3394          3504  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3395          3505  6                        INTMED_DATA<15, 1, 0> = 0;
;  3396          3506  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3397          3507  7                        IF .FLOAT_SCALE GTR (7 + .DESTINATION[DSC$B_SCALE])
;  3398          3508  6                        THEN
;  3399          3509  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3400          3510  6                        ELSE
;  3401          3511  7                            BEGIN
;  3402          3512  7                            TEMP_BUF1<6, 1, 0> = 1;
;  3403          3513  7                            TEMP_BUF1<0, 6, 0> = .INTMED_DATA<26, 6, 0>;
;  3404          3514  7                            FLOAT_SCALE = 7 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3405          3515  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3406          3516  8                                BEGIN
;  3407          3517  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3408          3518  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3409          3519  7                                END;
;  3410          3520  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3411          3521  7                            OUTPUT [BYTE_1] = .TEMP_BUF1 [S_BYTE_1];
;  3412          3522  6                            END;
;  3413          3523  5                        END;
;  3414          3524  4                END;
;  3415          3525  4
;  3416          3526  4            [DSC$K_DTYPE_WU]:
;  3417          3527  5                BEGIN
;  3418          3528  5
;  3419          3529  5                ! If the target is not a binary scale, then just move the
;  3420          3530  5                ! converted value in.
;  3421          3531  5                !
;  3422          3532  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3423          3533  5                THEN
;  3424          3534  6                    BEGIN
;  3425          3535  6                    IF .TEMP_BUF1 [LONG_1] GTRU K_LRGST_WU
;  3426          3536  6                    THEN
;  3427          3537  6                        SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  3428          3538  6                    OUTPUT [WORD_1] = .TEMP_BUF1 [WORD_1];
;  3429          3539  6                    END
;  3430          3540  5                ELSE
;  3431          3541  5
;  3432          3542  5                    ! If the sign and the scale of the H_float are zero,
;  3433          3543  5                    ! then the value is zero.
;  3434          3544  5                    !
;  3435          3545  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3436          3546  5                    THEN
;  3437          3547  5                        OUTPUT[WORD_1] = 0
;  3438          3548  5                    ELSE
;  3439          3549  6                        BEGIN
;  3440          3550  6                        TEMP_BUF1 = 0;
;  3441          3551  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3442          3552  6                        INTMED_DATA<15, 1, 0> = 0;
;  3443          3553  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3444          3554  7                        IF .FLOAT_SCALE GTR (15 + .DESTINATION[DSC$B_SCALE])
```

```
;  3445          3555  6                    THEN
;  3446          3556  6                        SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3447          3557  6                    ELSE
;  3448          3558  7                        BEGIN
;  3449          3559  7                        TEMP_BUF1<14, 1, 0> = 1;
;  3450          3560  7                        TEMP_BUF1<0, 14, 0> = .INTMED_DATA<18, 14, 0>;
;  3451          3561  7                        FLOAT_SCALE = 15 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3452          3562  7                        WHILE .FLOAT_SCALE GTR 0 DO
;  3453          3563  8                            BEGIN
;  3454          3564  8                            TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3455          3565  8                            FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3456          3566  7                            END;
;  3457          3567  7                        IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3458          3568  7                        OUTPUT [WORD_1] = .TEMP_BUF1 [S_WORD_1];
;  3459          3569  6                        END;
;  3460          3570  5                    END;
;  3461          3571  4                END;
;  3462          3572  4
;  3463          3573  4            [DSC$K_DTYPE_B]:
;  3464          3574  5                BEGIN
;  3465          3575  5
;  3466          3576  5                ! If the target is not a binary scale, then just move the
;  3467          3577  5                ! converted value in.
;  3468          3578  5                !
;  3469          3579  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3470          3580  5                THEN
;  3471          3581  5                    CVTLB (TEMP_BUF1, .OUTPUT)
;  3472          3582  5                ELSE
;  3473          3583  5
;  3474          3584  5                    ! If the sign and the scale of the H_Float are zero,
;  3475          3585  5                    ! then the value is zero.
;  3476          3586  5                    !
;  3477          3587  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3478          3588  5                    THEN
;  3479          3589  5                        OUTPUT[BYTE_1] = 0
;  3480          3590  5                    ELSE
;  3481          3591  6                        BEGIN
;  3482          3592  6                        TEMP_BUF1 = 0;
;  3483          3593  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3484          3594  6                        INTMED_DATA<15, 1, 0> = 0;
;  3485          3595  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3486          3596  7                        IF .FLOAT_SCALE GTR (7 + .DESTINATION[DSC$B_SCALE])
;  3487          3597  6                        THEN
;  3488          3598  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3489          3599  6                        ELSE
;  3490          3600  7                            BEGIN
;  3491          3601  7                            TEMP_BUF1<6, 1, 0> = 1;
;  3492          3602  7                            TEMP_BUF1<0, 6, 0> = .INTMED_DATA<26, 6, 0>;
;  3493          3603  7                            FLOAT_SCALE = 7 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3494          3604  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3495          3605  8                                BEGIN
;  3496          3606  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3497          3607  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3498          3608  7                                END;
;  3499          3609  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3500          3610  7                            OUTPUT [BYTE_1] = .TEMP_BUF1 [S_BYTE_1];
;  3501          3611  6                            END;
```

```
;  3502          3612  5                        END;
;  3503          3613  4                END;
;  3504          3614  4
;  3505          3615  4            [DSC$K_DTYPE_W]:
;  3506          3616  5                BEGIN
;  3507          3617  5
;  3508          3618  5                ! If the target is not a binary scale, then just move the
;  3509          3619  5                ! converted value in.
;  3510          3620  5                !
;  3511          3621  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3512          3622  5                THEN
;  3513          3623  5                    CVTLW (TEMP_BUF1, .OUTPUT)
;  3514          3624  5                ELSE
;  3515          3625  5
;  3516          3626  5                    ! If the sign and the scale of the H_Float are zero,
;  3517          3627  5                    ! then the value is zero.
;  3518          3628  5                    !
;  3519          3629  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3520          3630  5                    THEN
;  3521          3631  5                        OUTPUT[WORD_1] = 0
;  3522          3632  5                    ELSE
;  3523          3633  6                        BEGIN
;  3524          3634  6                        TEMP_BUF1 = 0;
;  3525          3635  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3526          3636  6                        INTMED_DATA<15, 1, 0> = 0;
;  3527          3637  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3528          3638  7                        IF .FLOAT_SCALE GTR (15 + .DESTINATION[DSC$B_SCALE])
;  3529          3639  6                        THEN
;  3530          3640  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3531          3641  6                        ELSE
;  3532          3642  7                            BEGIN
;  3533          3643  7                            TEMP_BUF1<14, 1, 0> = 1;
;  3534          3644  7                            TEMP_BUF1<0, 14, 0> = .INTMED_DATA<18, 14, 0>;
;  3535          3645  7                            FLOAT_SCALE = 15 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3536          3646  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3537          3647  8                                BEGIN
;  3538          3648  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3539          3649  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3540          3650  7                                END;
;  3541          3651  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3542          3652  7                            OUTPUT [WORD_1] = .TEMP_BUF1 [S_WORD_1];
;  3543          3653  6                            END;
;  3544          3654  5                        END;
;  3545          3655  4                END;
;  3546          3656  4
;  3547          3657  4            [DSC$K_DTYPE_L]:
;  3548          3658  5                BEGIN
;  3549          3659  5
;  3550          3660  5                ! If the target is not a binary scale, then just move the
;  3551          3661  5                ! converted value in.
;  3552          3662  5                !
;  3553          3663  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3554          3664  5                THEN
;  3555          3665  5                    OUTPUT [LONG_1] = .TEMP_BUF1 [S_LONG_1]
;  3556          3666  5                ELSE
;  3557          3667  5
;  3558          3668  5                    ! If the sign and the scale of the H_Float are zero,
```

```
;  3559          3669  5                    ! then the value is zero.
;  3560          3670  5                    !
;  3561          3671  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3562          3672  5                    THEN
;  3563          3673  5                        OUTPUT[LONG_1] = 0
;  3564          3674  5                    ELSE
;  3565          3675  6                        BEGIN
;  3566          3676  6                        TEMP_BUF1 = 0;
;  3567          3677  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3568          3678  6                        INTMED_DATA<15, 1, 0> = 0;
;  3569          3679  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3570          3680  7                        IF .FLOAT_SCALE GTR (31 + .DESTINATION[DSC$B_SCALE])
;  3571          3681  6                        THEN
;  3572          3682  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3573          3683  6                        ELSE
;  3574          3684  7                            BEGIN
;  3575          3685  7                            TEMP_BUF1<30, 1, 0> = 1;
;  3576          3686  7                            TEMP_BUF1<14, 16, 0> = .INTMED_DATA<16, 16, 0>;
;  3577          3687  7                            TEMP_BUF1<0, 14, 0> = .(INTMED_DATA+4)<18, 14, 0>;
;  3578          3688  7                            FLOAT_SCALE = 31 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3579          3689  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3580          3690  8                                BEGIN
;  3581          3691  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3582          3692  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3583          3693  7                                END;
;  3584          3694  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3585          3695  7                            OUTPUT [LONG_1] = .TEMP_BUF1 [S_LONG_1];
;  3586          3696  6                            END;
;  3587          3697  5                        END;
;  3588          3698  4                END;
;  3589          3699  4
;  3590          3700  4                [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  3591          3701  5                    BEGIN
;  3592          3702  5                    MAP
;  3593          3703  5                        OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  3594          3704  5                        INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  3595          3705  5
;  3596          3706  5                    INCR I FROM 0 TO .DST_INFO[D_LEN] - 1 DO
;  3597          3707  6                        BEGIN
;  3598          3708  6                        OUTPUT[.I] = .INTMED_DATA[.I];
;  3599          3709  5                        END;
;  3600          3710  4                    END;
;  3601          3711  4
;  3602          3712  4                [INRANGE, OUTRANGE]:
;  3603          3713  4                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_smlint');
;  3604          3714  4                TES;                                !For DEC_SMLINT
;  3605          3715  3            END;
```

```
;  3607          3716  3
;  3608          3717  3          [K_DEC_LRGINT]:
;  3609          3718  4              BEGIN
;  3610          3719  4              M_SCALE_P_P;
;  3611          3720  4
;  3612          3721  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_LU TO DSC$K_DTYPE_0 OF
;  3613          3722  4                  SET
;  3614          3723  4
;  3615          3724  4                  [DSC$K_DTYPE_LU]:
;  3616          3725  5                      BEGIN
;  3617          3726  6                      IF (CMPP (NO_DIGITS, INTMED_DATA, %REF (K_PACK_LU_LEN), .LRGST_P_LU) GEQ 0)
;  3618          3727  5                      THEN
;  3619          3728  5                          SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  3620          3729  5                      BICPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3621          3730  5                      CVTPL (NO_DIGITS, INTMED_DATA, .OUTPUT);
;  3622          3731  5                      BISPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3623          3732  4                      END;
;  3624          3733  4
;  3625          3734  4                  [DSC$K_DTYPE_Q, DSC$K_DTYPE_QU, DSC$K_DTYPE_O]:
;  3626          3735  5                      BEGIN
;  3627          3736  5                      CVTPS (NO_DIGITS, INTMED_DATA, NO_DIGITS, TEMP_BUF2);
;  3628          3737  5                      CLASS_S_DESC [DSC$W_LENGTH] = .NO_DIGITS + 1;
;  3629          3738  5                      CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  3630          3739  5                      OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF1);
;  3631          3740  5                      IF .DESTINATION[DSC$B_DTYPE] EQL DSC$K_DTYPE_Q OR
;  3632          3741  5                         .DESTINATION[DSC$B_DTYPE] EQL DSC$K_DTYPE_QU
;  3633          3742  5                      THEN
;  3634          3743  5                          CVTRHQ (TEMP_BUF1, .OUTPUT)
;  3635          3744  5                      ELSE
;  3636          3745  5                          CVTRHO (TEMP_BUF1, OUTPUT);
;  3637          3746  4                      END;
;  3638          3747  4
;  3639          3748  4                  [INRANGE, OUTRANGE]:
;  3640          3749  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  dec_lrgint');
;  3641          3750  4                  TES;                                !For DEC_LRGINT
;  3642          3751  3          END;
```

```
;  3644          3752  3
;  3645          3753  3          [K_NBDS_SMLINT]:
;  3646          3754  4              BEGIN
;  3647          3755  4              CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  3648          3756  4              CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  3649          3757  4              STATUS = OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  3650          3758  4                  (K_IGN_BLKS_OR_K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  3651          3759  4              IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  3652          3760  4
;  3653          3761  4              ! This is a HACK for scaled binary.  If the destination is Scaled
;  3654          3762  4              ! Binary we will leave the value as a H_Float so that we can
;  3655          3763  4              ! do the convert to Scaled Binary by hand.  The algorithm follows:
;  3656          3764  4              !
;  3657          3765  4              ! This is the algorithm for the code in the particular case below:
;  3658          3766  4              ! 1) Get the sign.
;  3659          3767  4              ! 2) Get the scale of the H_Float.
;  3660          3768  4              ! 3) Check if an overflow will occur.  An underflow is
;  3661          3769  4              !       acceptable and will be truncated automatically.
;  3662          3770  4              ! 4) Move the most significant H_Float fractional bits
;  3663          3771  4              !       into the temporary destination.
;  3664          3772  4              !       (Note: this includes the redundant most significant
;  3665          3773  4              !       fraction bit.
;  3666          3774  4              ! 5) Alter the destination to the correct scale.
;  3667          3775  4              ! 6) This is an absolute value so correct for the sign.
;  3668          3776  4              ! 7) Move the result into the final destination.
;  3669          3777  4              !
;  3670          3778  4              ! ****************** HACK - BAB Dec. 1983 ********************
;  3671          3779  4              !
;  3672          3780  4              IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3673          3781  4              THEN
;  3674          3782  4                  IF .CVT_ROUND_FLAG
;  3675          3783  4                  THEN
;  3676          3784  4                      CVTRHL (TEMP_BUF1, TEMP_BUF2)
;  3677          3785  4                  ELSE
;  3678          3786  4                      CVTHL  (TEMP_BUF1, TEMP_BUF2);
;  3679          3787  4
;  3680          3788  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_V TO DSC$K_DTYPE_SVU OF
;  3681          3789  4                  SET
;  3682          3790  4
;  3683          3791  4                  [DSC$K_DTYPE_BU]:
;  3684          3792  5                      BEGIN
;  3685          3793  5
;  3686          3794  5                      ! If the target is not a binary scale, then just move the
;  3687          3795  5                      ! converted value in.
;  3688          3796  5                      !
;  3689          3797  5                      IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3690          3798  5                      THEN
;  3691          3799  6                          BEGIN
;  3692          3800  6                          IF .TEMP_BUF2 [LONG_1] GTRU K_LRGST_BU
;  3693          3801  6                          THEN
;  3694          3802  6                              SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  3695          3803  6                          OUTPUT [BYTE_1] = .TEMP_BUF2 [BYTE_1];
;  3696          3804  6                          END
;  3697          3805  5                      ELSE
;  3698          3806  5
;  3699          3807  5                          ! If the sign and the scale of the H_Float are zero,
;  3700          3808  5                          ! then the value is zero.
```

```
;  3701          3809  5                    !
;  3702          3810  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3703          3811  5                    THEN
;  3704          3812  5                        OUTPUT[BYTE_1] = 0
;  3705          3813  5                    ELSE
;  3706          3814  6                        BEGIN
;  3707          3815  6                        TEMP_BUF1 = 0;
;  3708          3816  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3709          3817  6                        INTMED_DATA<15, 1, 0> = 0;
;  3710          3818  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3711          3819  7                        IF .FLOAT_SCALE GTR (7 + .DESTINATION[DSC$B_SCALE])
;  3712          3820  6                        THEN
;  3713          3821  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3714          3822  6                        ELSE
;  3715          3823  7                            BEGIN
;  3716          3824  7                            TEMP_BUF1<6, 1, 0> = 1;
;  3717          3825  7                            TEMP_BUF1<0, 6, 0> = .INTMED_DATA<26, 6, 0>;
;  3718          3826  7                            FLOAT_SCALE = 7 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3719          3827  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3720          3828  8                                BEGIN
;  3721          3829  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3722          3830  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3723          3831  7                                END;
;  3724          3832  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3725          3833  7                            OUTPUT [BYTE_1] = .TEMP_BUF1 [S_BYTE_1];
;  3726          3834  6                            END;
;  3727          3835  5                        END;
;  3728          3836  4                END;
;  3729          3837  4
;  3730          3838  4            [DSC$K_DTYPE_WU]:
;  3731          3839  5                BEGIN
;  3732          3840  5
;  3733          3841  5                ! If the target is not a binary scale, then just move the
;  3734          3842  5                ! converted value in.
;  3735          3843  5                !
;  3736          3844  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3737          3845  5                THEN
;  3738          3846  6                    BEGIN
;  3739          3847  6                    IF .TEMP_BUF2 [LONG_1] GTRU K_LRGST_WU
;  3740          3848  6                    THEN
;  3741          3849  6                        SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  3742          3850  6                    OUTPUT [WORD_1] = .TEMP_BUF2 [WORD_1];
;  3743          3851  6                    END
;  3744          3852  5                ELSE
;  3745          3853  5
;  3746          3854  5                    ! If the sign and the scale of the H_Float are zero,
;  3747          3855  5                    ! then the value is zero.
;  3748          3856  5                    !
;  3749          3857  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3750          3858  5                    THEN
;  3751          3859  5                        OUTPUT[WORD_1] = 0
;  3752          3860  5                    ELSE
;  3753          3861  6                        BEGIN
;  3754          3862  6                        TEMP_BUF1 = 0;
;  3755          3863  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3756          3864  6                        INTMED_DATA<15, 1, 0> = 0;
;  3757          3865  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
```

```
;  3758     3866  7                    IF .FLOAT_SCALE GTR (15 + .DESTINATION[DSC$B_SCALE])
;  3759     3867  6                    THEN
;  3760     3868  6                        SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3761     3869  6                    ELSE
;  3762     3870  7                        BEGIN
;  3763     3871  7                        TEMP_BUF1<14, 1, 0> = 1;
;  3764     3872  7                        TEMP_BUF1<0, 14, 0> = .INTMED_DATA<18, 14, 0>;
;  3765     3873  7                        FLOAT_SCALE = 15 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3766     3874  7                        WHILE .FLOAT_SCALE GTR 0 DO
;  3767     3875  8                            BEGIN
;  3768     3876  8                            TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3769     3877  8                            FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3770     3878  7                            END;
;  3771     3879  7                        IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3772     3880  7                        OUTPUT [WORD_1] = .TEMP_BUF1 [S_WORD_1];
;  3773     3881  6                        END;
;  3774     3882  5                    END;
;  3775     3883  4                END;
;  3776     3884  4
;  3777     3885  4            [DSC$K_DTYPE_B]:
;  3778     3886  5                BEGIN
;  3779     3887  5
;  3780     3888  5                ! If the target is not a binary scale, then just move the
;  3781     3889  5                ! converted value in.
;  3782     3890  5                !
;  3783     3891  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3784     3892  5                THEN
;  3785     3893  5                    CVTLB (TEMP_BUF2, .OUTPUT)
;  3786     3894  5                ELSE
;  3787     3895  5
;  3788     3896  5                    ! If the sign and the scale of the H_Float are zero,
;  3789     3897  5                    ! then the value is zero.
;  3790     3898  5                    !
;  3791     3899  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3792     3900  5                    THEN
;  3793     3901  5                        OUTPUT[BYTE_1] = 0
;  3794     3902  5                    ELSE
;  3795     3903  6                        BEGIN
;  3796     3904  6                        TEMP_BUF1 = 0;
;  3797     3905  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3798     3906  6                        INTMED_DATA<15, 1, 0> = 0;
;  3799     3907  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3800     3908  7                        IF .FLOAT_SCALE GTR (7 + .DESTINATION[DSC$B_SCALE])
;  3801     3909  6                        THEN
;  3802     3910  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3803     3911  6                        ELSE
;  3804     3912  7                            BEGIN
;  3805     3913  7                            TEMP_BUF1<6, 1, 0> = 1;
;  3806     3914  7                            TEMP_BUF1<0, 6, 0> = .INTMED_DATA<26, 6, 0>;
;  3807     3915  7                            FLOAT_SCALE = 7 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3808     3916  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3809     3917  8                                BEGIN
;  3810     3918  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3811     3919  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3812     3920  7                                END;
;  3813     3921  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3814     3922  7                            OUTPUT [BYTE_1] = .TEMP_BUF1 [S_BYTE_1];
```

```
;  3815     3923  6                        END;
;  3816     3924  5                    END;
;  3817     3925  4                END;
;  3818     3926  4
;  3819     3927  4            [DSC$K_DTYPE_W]:
;  3820     3928  4                BEGIN
;  3821     3929  4
;  3822     3930  5                ! If the target is not a binary scale, then just move the
;  3823     3931  5                ! converted value in.
;  3824     3932  5                !
;  3825     3933  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3826     3934  5                THEN
;  3827     3935  5                    CVTLW (TEMP_BUF2, .OUTPUT)
;  3828     3936  5                ELSE
;  3829     3937  5
;  3830     3938  5                    ! If the sign and the scale of the H_Float are zero,
;  3831     3939  5                    ! then the value is zero.
;  3832     3940  5                    !
;  3833     3941  5                    IF .INTMED_DATA[WORD_1] EQL 0
;  3834     3942  5                    THEN
;  3835     3943  5                        OUTPUT[WORD_1] = 0
;  3836     3944  5                    ELSE
;  3837     3945  6                        BEGIN
;  3838     3946  6                        TEMP_BUF1 = 0;
;  3839     3947  6                        SIGN = .INTMED_DATA<15, 1, 0>;
;  3840     3948  6                        INTMED_DATA<15, 1, 0> = 0;
;  3841     3949  6                        FLOAT_SCALE = .INTMED_DATA[WORD_1] - 16384;
;  3842     3950  7                        IF .FLOAT_SCALE GTR (15 + .DESTINATION[DSC$B_SCALE])
;  3843     3951  6                        THEN
;  3844     3952  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3845     3953  6                        ELSE
;  3846     3954  7                            BEGIN
;  3847     3955  7                            TEMP_BUF1<14, 1, 0> = 1;
;  3848     3956  7                            TEMP_BUF1<0, 14, 0> = .INTMED_DATA<18, 14, 0>;
;  3849     3957  7                            FLOAT_SCALE = 15 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3850     3958  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3851     3959  8                                BEGIN
;  3852     3960  8                                TEMP_BUF1[LONG_1] = .TEMP_BUF1[S_LONG_1] / 2;
;  3853     3961  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3854     3962  7                                END;
;  3855     3963  7                            IF .SIGN THEN TEMP_BUF1 = 0 - .TEMP_BUF1;
;  3856     3964  7                            OUTPUT [WORD_1] = .TEMP_BUF1 [S_WORD_1];
;  3857     3965  6                            END;
;  3858     3966  5                        END;
;  3859     3967  4                END;
;  3860     3968  4
;  3861     3969  4            [DSC$K_DTYPE_L]:
;  3862     3970  5                BEGIN
;  3863     3971  5
;  3864     3972  5                ! If the target is not a binary scale, then just move the
;  3865     3973  5                ! converted value in.
;  3866     3974  5                !
;  3867     3975  5                IF NOT .DESTINATION [DSC$V_FL_BINSCALE]
;  3868     3976  5                THEN
;  3869     3977  5                    OUTPUT [LONG_1] = .TEMP_BUF2 [S_LONG_1]
;  3870     3978  5                ELSE
;  3871     3979  5
```

```
;  3872          3980  5                    ! If the sign and the scale of the D_float are zero,
;  3873          3981  5                    ! then the value is zero.
;  3874          3982  5                    !
;  3875          3983  5                    IF .TEMP_BUF1[WORD_1] EQL 0
;  3876          3984  5                    THEN
;  3877          3985  5                        OUTPUT[LONG_1] = 0
;  3878          3986  5                    ELSE
;  3879          3987  6                        BEGIN
;  3880          3988  6                        TEMP_BUF2 = 0;
;  3881          3989  6                        SIGN = .TEMP_BUF1<15, 1, 0>;
;  3882          3990  6                        TEMP_BUF1<15, 1, 0> = 0;
;  3883          3991  6                        FLOAT_SCALE = .TEMP_BUF1[WORD_1] - 16384;
;  3884          3992  7                        IF .FLOAT_SCALE GTR (31 + .DESTINATION[DSC$B_SCALE])
;  3885          3993  6                        THEN
;  3886          3994  6                            SIGNAL(DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME)
;  3887          3995  6                        ELSE
;  3888          3996  7                            BEGIN
;  3889          3997  7                            TEMP_BUF2<30, 1, 0> = 1;
;  3890          3998  7                            TEMP_BUF2<14, 16, 0> = .TEMP_BUF1<16, 16, 0>;
;  3891          3999  7                            TEMP_BUF2<0, 14, 0> = .(TEMP_BUF1+4)<18, 14, 0>;
;  3892          4000  7                            FLOAT_SCALE = 31 + .DESTINATION[DSC$B_SCALE] - .FLOAT_SCALE;
;  3893          4001  7                            WHILE .FLOAT_SCALE GTR 0 DO
;  3894          4002  8                                BEGIN
;  3895          4003  8                                TEMP_BUF2[LONG_1] = .TEMP_BUF2[S_LONG_1] / 2;
;  3896          4004  8                                FLOAT_SCALE = .FLOAT_SCALE - 1;
;  3897          4005  7                                END;
;  3898          4006  7                            IF .SIGN THEN TEMP_BUF2 = 0 - .TEMP_BUF2;
;  3899          4007  7                            OUTPUT [LONG_1] = .TEMP_BUF2 [S_LONG_1];
;  3900          4008  6                            END;
;  3901          4009  5                        END;
;  3902          4010  4                END;
;  3903          4011  4
;  3904          4012  4            [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  3905          4013  5                BEGIN
;  3906          4014  5                MAP
;  3907          4015  5                    OUTPUT:  REF BITVECTOR[K_OUTPUT_BUFFER_LENGTH * 8],
;  3908          4016  5                    INTMED_DATA:  BITVECTOR[K_INTMED_DATA_LENGTH * 8];
;  3909          4017  5
;  3910          4018  5                INCR I FROM 0 TO .DST_INFO[D_LEN] - 1 DO
;  3911          4019  6                    BEGIN
;  3912          4020  6                    OUTPUT[.I] = .INTMED_DATA[.I];
;  3913          4021  5                    END;
;  3914          4022  4                END;
;  3915          4023  4
;  3916          4024  4            [INRANGE, OUTRANGE]:
;  3917          4025  4                $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_smlint');
;  3918          4026  4            TES;                                !For NBDS_SMLINT
;  3919          4027  3        END;
```

```
;  3921          4028  3          [K_NBDS_LRGINT]:
;  3922          4029  3              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_LU TO DSC$K_DTYPE_O OF
;  3923          4030  3                  SET
;  3924          4031  3
;  3925          4032  3                  [DSC$K_DTYPE_LU]:
;  3926          4033  3                      BEGIN
;  3927          4034  4                      CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  3928          4035  4                      CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  3929          4036  4                      STATUS = OTS$CVT_T_D (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  3930          4037  4                          (K_IGN_BLKS_OR_K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  3931          4038  4                      IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  3932          4039  4                      IF .TEMP_BUF1<15, 1, 0> THEN SIGNAL (DBG$_CVTNEGUNS, 1, .DBG$GL_OPCODE_NAME);
;  3933          4040  4                      IF CMPD (TEMP_BUF1, .LRGST_D_LU) GTR 0 THEN SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME
;  3934          4041  4                      BICPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3935          4042  4                      IF .CVT_ROUND_FLAG
;  3936          4043  4                      THEN
;  3937          4044  4                          CVTRDL (TEMP_BUF1, .OUTPUT)
;  3938          4045  4                      ELSE
;  3939          4046  4                          CVTDL  (TEMP_BUF1, .OUTPUT);
;  3940          4047  4
;  3941          4048  4                      BISPSW (%REF (K_SET_ARITHMETIC_TRAP));
;  3942          4049  4                      END;
;  3943          4050  3
;  3944          4051  3                  [DSC$K_DTYPE_Q, DSC$K_DTYPE_QU]:                                    ! M003
;  3945          4052  3                      BEGIN
;  3946          4053  4                      CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  3947          4054  4                      CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  3948          4055  4                      STATUS = OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  3949          4056  4                          (K_IGN_BLKS_OR_K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  3950          4057  4                      IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  3951          4058  4                      CVTRHQ (TEMP_BUF1, .OUTPUT)
;  3952          4059  4                      END;
;  3953          4060  3
;  3954          4061  3                  [DSC$K_DTYPE_O]:                                                    ! A004
;  3955          4062  3                      BEGIN                                                           ! A004
;  3956          4063  4                      LOCAL                                                           ! A004
;  3957          4064  4                          Sign_flag : INITIAL( 0 ),                                   ! A004
;  3958          4065  4                          Current_character;                                          ! A004
;  3959          4066  4                      MAP OUTPUT : REF VECTOR[4];                                     ! A004
;  3960          4067  4                      !++                                                             ! A004
;  3961          4068  4                      ! Init the octaword                                             ! A004
;  3962          4069  4                      !--                                                             ! A004
;  3963          4070  4                      OUTPUT[ 3 ] = 0;                                                ! A004
;  3964          4071  4                      OUTPUT[ 2 ] = 0;                                                ! A004
;  3965          4072  4                      OUTPUT[ 1 ] = 0;                                                ! A004
;  3966          4073  4                      OUTPUT[ 0 ] = CH$RCHAR( .SRC_INFO[ S_POINTER ] ) - %C'0';       ! A004
;  3967          4074  4                      !++                                                             ! A004
;  3968          4075  4                      ! Test for bad characters                                       ! A004
;  3969          4076  4                      !--                                                             ! A004
;  3970          4077  4                      IF .OUTPUT[ 0 ] LSS 0  OR  .OUTPUT[ 0 ] GTR 9                   ! A004
;  3971          4078  4                      THEN                                                            ! A004
;  3972          4079  4                          IF .OUTPUT[ 0 ] EQL %C'-' - %C'0'                           ! A004
;  3973          4080  4                          THEN                                                        ! A004
;  3974          4081  4
;  3975          4082  4
;  3976          4083  4
;  3977          4084  4
```

```
; 3978        4085  5                BEGIN                                                                  ! A004
; 3979        4086  5                SRC_INFO[ S_POINTER ] = .SRC_INFO[ S_POINTER ] + 1;                    ! A004
; 3980        4087  5                SRC_INFO[ S_LEN ] = .SRC_INFO[ S_LEN ] - 1;                            ! A004
; 3981        4088  5                OUTPUT[ 0 ] = CH$RCHAR( .SRC_INFO[ S_POINTER ] ) -                     ! A004
; 3982        4089  5                                  %C'0';                                               ! A004
; 3983        4090  5                Sign_flag = 1;                                                         ! A004
; 3984        4091  5                END                                                                    ! A004
; 3985        4092  4            ELSE                                                                       ! A004
; 3986        4093  4                SIGNAL( DBG$_INVDIGDEC, 2, 1, .SRC_INFO[S_POINTER]);                   ! A004
; 3987        4094  4
; 3988        4095  4            !++                                                                        ! A004
; 3989        4096  4            ! For each character                                                       ! A004
; 3990        4097  4            !    multiply by 10                                                        ! A004
; 3991        4098  4            !    add it to the low order long word                                     ! A004
; 3992        4099  4            !--                                                                        ! A004
; 3993        4100  4            INCR Current_char_num FROM 1 TO .SRC_INFO [S_LEN] - 1 DO                   ! A004
; 3994        4101  5                BEGIN                                                                  ! A004
; 3995        4102  5                DBG$CVT_SCALE_OU_UP_BY_10_R1( .OUTPUT );                               ! A004
; 3996        4103  5                Current_character = CH$RCHAR( .SRC_INFO[ S_POINTER ] +                 ! A004
; 3997        4104  5                                              .Current_char_num ) -                     ! A004
; 3998        4105  5                                  %C'0';                                               ! A004
; 3999        4106  5
; 4000        4107  5                !++                                                                    ! A004
; 4001        4108  5                ! Test for bad characters                                              ! A004
; 4002        4109  5                !--                                                                    ! A004
; 4003        4110  5                IF (.Current_character LSS 0)  OR                                      ! A004
; 4004        4111  6                   (.Current_character GTR 9)                                          ! A004
; 4005        4112  5                THEN                                                                   ! A004
; 4006        4113  5                    SIGNAL( DBG$_INVDIGDEC, 2, 1, .SRC_INFO[S_POINTER] +               ! A004
; 4007        4114  5                                                  .Current_char_num );                  ! A004
; 4008        4115  5
; 4009        4116  5                OUTPUT[ 0 ] = .OUTPUT[ 0 ] + .Current_character;                       ! A004
; 4010        4117  4                END;                                                                   ! A004
; 4011        4118  4
; 4012        4119  4                !++
; 4013        4120  4                ! When there was a negative we subtract from 0
; 4014        4121  4                !--
; 4015        4122  4                IF .Sign_flag                                                          ! A004
; 4016        4123  4                THEN                                                                   ! A004
; 4017        4124  5                    BEGIN                                                              ! A004
; 4018        4125  5                    LOCAL                                                              ! A004
; 4019        4126  5                        Octaword_zero : VECTOR[4];                                     ! A004
; 4020        4127  5
; 4021        4128  5                    Octaword_zero[ 3 ] = 0;                                            ! A004
; 4022        4129  5                    Octaword_zero[ 2 ] = 0;                                            ! A004
; 4023        4130  5                    Octaword_zero[ 1 ] = 0;                                            ! A004
; 4024        4131  5                    Octaword_zero[ 0 ] = 0;                                            ! A004
; 4025        4132  5
; 4026        4133  5                    SUBM( 4, .OUTPUT, Octaword_zero, .OUTPUT );                        ! A004
; 4027        4134  5
; 4028        4135  4                    END;                                                               ! A004
; 4029        4136  4
; 4030        4137  3            END;                                                                       ! A004
; 4031        4138  3
; 4032        4139  3        [INRANGE, OUTRANGE]:
; 4033        4140  3            $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_lrgint');
; 4034        4141  3        TES;                                !For NBDS_LRGINT
```

```
;  4036          4142  3
;  4037          4143  3          [K_NBDS_LRGFLTCMPLX]:
;  4038          4144  4              BEGIN
;  4039          4145  4              CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  4040          4146  4              CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  4041          4147  4              CASE .DESTINATION [DSC$B_DTYPE] FROM DSC$K_DTYPE_G TO DSC$K_DTYPE_HC OF
;  4042          4148  4                  SET
;  4043          4149  4
;  4044          4150  4                  [DSC$K_DTYPE_G, DSC$K_DTYPE_GC]:
;  4045          4151  5                      BEGIN
;  4046          4152  5                      STATUS = OTS$CVT_T_G (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  4047          4153  5                          (K_IGN_BLKS OR K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  4048          4154  5                      IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  4049          4155  5                      OUTPUT [LONG_1] = .TEMP_BUF1 [LONG_1];
;  4050          4156  5                      OUTPUT [LONG_2] = .TEMP_BUF1 [LONG_2];
;  4051          4157  5                      IF .DESTINATION[DSC$B_DTYPE] EQL DSC$K_DTYPE_GC
;  4052          4158  5                      THEN
;  4053          4159  6                          BEGIN
;  4054          4160  6
;  4055          4161  6                          ! Fill in imaginary part with 0;
;  4056          4162  6                          !
;  4057          4163  6                          OUTPUT[LONG_3] = 0;
;  4058          4164  6                          OUTPUT[LONG_4] = 0;
;  4059          4165  6                          END;
;  4060          4166  5                      END;
;  4061          4167  4
;  4062          4168  4                  [DSC$K_DTYPE_H, DSC$K_DTYPE_HC]:
;  4063          4169  4                      BEGIN
;  4064          4170  5                      STATUS = OTS$CVT_T_H (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  4065          4171  5                          (K_IGN_BLKS OR K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  4066          4172  5                      IF NOT .STATUS THEN SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
;  4067          4173  5                      CH$MOVE (16, TEMP_BUF1, .OUTPUT);
;  4068          4174  5                      IF .DESTINATION[DSC$B_DTYPE] EQL DSC$K_DTYPE_HC
;  4069          4175  5                      THEN
;  4070          4176  5
;  4071          4177  5                          ! Fill in imaginary part with 0.
;  4072          4178  5                          !
;  4073          4179  5                          CH$FILL (0, 16, .OUTPUT+16);
;  4074          4180  5                      END;
;  4075          4181  4
;  4076          4182  4                  [INRANGE, OUTRANGE]:
;  4077          4183  4                      $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_lrgfltcmplx');
;  4078          4184  4                  TES;                                !For NBDS_LRGFLTCMPLX
;  4079          4185  4          END;
```

```
;  4083          4188  3
;  4084          4189  3          [K_NBDS_NBDS]:
;  4085          4190  3              SELECTONE .DESTINATION [DSC$B_DTYPE] OF
;  4086          4191  3                  SET
;  4087          4192  3
;  4088          4193  3                  [DSC$K_DTYPE_BU, DSC$K_DTYPE_T, DSC$K_DTYPE_VT, DSC$K_DTYPE_AC, DSC$K_DTYPE_AZ]:
;  4089          4194  4                      BEGIN
;  4090          4195  4                      IF .SCALE NEQ 0
;  4091          4196  4                      THEN
;  4092          4197  5                          BEGIN
;  4093          4198  5                          CLASS_S_DESC [DSC$W_LENGTH] = .SRC_INFO [S_LEN];
;  4094          4199  5                          CLASS_S_DESC [DSC$A_POINTER] = .SRC_INFO [S_POINTER];
;  4095          4200  5                          STATUS = OTS$CVT_T_A (CLASS_S_DESC, TEMP_BUF1, 0, -.SCALE,
;  4096          4201  5                              (K_IGN_BLKS OR K_ENB_UNDERFLOW OR K_IGN_TABS OR K_ENB_SCALE));
;  4097          4202  5                          IF .STATUS
;  4098          4203  5                          THEN
;  4099          4204  6                              BEGIN
;  4100          4205  6                              CLASS_S_DESC [DSC$W_LENGTH] = K_TEMP_BUF_LENGTH;
;  4101          4206  6                              CLASS_S_DESC [DSC$A_POINTER] = TEMP_BUF2;
;  4102          4207  6                              IF .DST_INFO [D_LEN] - 9 LEQ 0
;  4103          4208  6                              THEN
;  4104          4209  6                                  DIGITS_IN_FRACT = 33
;  4105          4210  6                              ELSE
;  4106          4211  6                                  DIGITS_IN_FRACT = MIN (33, .DST_INFO [D_LEN] - 9);
;  4107          4212  6                              STATUS = FOR$CVT_H_TE (TEMP_BUF1, CLASS_S_DESC, .DIGITS_IN_FRACT, 0, 0, 4);
;  4108          4213  6                              IF NOT .STATUS THEN $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  error in h-to-te conve
;  4109          4214  6                              BUF_OFFSET = CH$FIND_NOT_CH (K_TEMP_BUF_LENGTH, TEMP_BUF2, %C' ') - TEMP_BUF2;
;  4110          4215  6                              FINAL_LEN = K_TEMP_BUF_LENGTH - .BUF_OFFSET;
;  4111          4216  6                              OUTPUT_STR_LEN = .FINAL_LEN;
;  4112          4217  6
;  4113          4218  6                              SELECTONE .DESTINATION[DSC$B_DTYPE] OF
;  4114          4219  6                                  SET
;  4115          4220  6                                  [DSC$K_DTYPE_AC]:
;  4116          4221  7                                      BEGIN
;  4117          4222  7                                      MAP
;  4118          4223  7                                          OUTPUT:  REF VECTOR[, BYTE];
;  4119          4224  7                                      CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  4120          4225  7                                      CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[1];
;  4121          4226  7                                      STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_D
;  4122          4227  7                                      IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, T, .DBG$GL_OPCODE_
;  4123          4228  7                                      IF NOT .STATUS THEN SIGNAL (.STATUS);
;  4124          4229  7                                      OUTPUT[0] = .FINAL_LEN;
;  4125          4230  6                                      END;
;  4126          4231  6
;  4127          4232  6                                  [DSC$K_DTYPE_AZ]:
;  4128          4233  7                                      BEGIN
;  4129          4234  7                                      MAP
;  4130          4235  7                                          OUTPUT:  REF VECTOR[, BYTE];
;  4131          4236  7                                      CLASS_S_DESC[DSC$W_LENGTH] = .FINAL_LEN;
;  4132          4237  7                                      CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[0];
;  4133          4238  7                                      STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, CLASS_S_D
;  4134          4239  7                                      IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, T, .DBG$GL_OPCODE_
;  4135          4240  7                                      IF NOT .STATUS THEN SIGNAL (.STATUS);
;  4136          4241  7                                      OUTPUT[.FINAL_LEN + 1] = 0;
;  4137          4242  6                                      END;
;  4138          4243  6
;  4139          4244  6                                  [OTHERWISE]:
```

```
; 4140          4245  7                                 BEGIN
; 4141          4246  7                                 STATUS = LIB$SCOPY_R_DX6 (.FINAL_LEN, TEMP_BUF2 + .BUF_OFFSET, .DESTINAT
; 4142          4247  7                                 IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, T, .DBG$GL_OPCODE_
; 4143          4248  7                                 IF NOT .STATUS THEN SIGNAL (.STATUS);
; 4144          4249  6                                 END;
; 4145          4250  6                             TES;
; 4146          4251  6                         END
; 4147          4252  5                     ELSE
; 4148          4253  5                         SIGNAL (DBG$_INVNUMSTR, 1, .DBG$GL_OPCODE_NAME);
; 4149          4254  5                     END
; 4150          4255  4                 ELSE
; 4151          4256  5                     BEGIN
; 4152          4257  5                     OUTPUT_STR_LEN = .SOURCE [DSC$W_LENGTH];
; 4153          4258  5                     SELECTONE .DESTINATION[DSC$B_DTYPE] OF
; 4154          4259  5                         SET
; 4155          4260  5                         [DSC$K_DTYPE_AC]:
; 4156          4261  6                             BEGIN
; 4157          4262  6                             MAP
; 4158          4263  6                                 OUTPUT:  REF VECTOR[, BYTE];
; 4159          4264  6                             CLASS_S_DESC[DSC$W_LENGTH] = .SOURCE[DSC$W_LENGTH];
; 4160          4265  6                             CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[1];
; 4161          4266  6                             STATUS = LIB$SCOPY_DXDX6 (.SOURCE, CLASS_S_DESC);
; 4162          4267  6                             IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME
; 4163          4268  6                             IF NOT .STATUS THEN SIGNAL (.STATUS);
; 4164          4269  6                             OUTPUT[0] = .SOURCE[DSC$W_LENGTH];
; 4165          4270  5                             END;
; 4166          4271  5
; 4167          4272  5                         [DSC$K_DTYPE_AZ]:
; 4168          4273  6                             BEGIN
; 4169          4274  6                             MAP
; 4170          4275  6                                 OUTPUT:  REF VECTOR[, BYTE];
; 4171          4276  6                             CLASS_S_DESC[DSC$W_LENGTH] = .SOURCE[DSC$W_LENGTH];
; 4172          4277  6                             CLASS_S_DESC[DSC$A_POINTER] = OUTPUT[0];
; 4173          4278  6                             STATUS = LIB$SCOPY_DXDX6 (.SOURCE, CLASS_S_DESC);
; 4174          4279  6                             IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME
; 4175          4280  6                             IF NOT .STATUS THEN SIGNAL (.STATUS);
; 4176          4281  6                             OUTPUT[.SOURCE[DSC$W_LENGTH]] = 0;
; 4177          4282  5                             END;
; 4178          4283  5
; 4179          4284  5                         [OTHERWISE]:
; 4180          4285  6                             BEGIN
; 4181          4286  6                             STATUS = LIB$SCOPY_DXDX6 (.SOURCE, .DESTINATION);
; 4182          4287  6                             IF .STATUS EQL LIB$_STRTRU THEN SIGNAL (DBG$_ISTRTRU, 1, .DBG$GL_OPCODE_NAME
; 4183          4288  6                             IF NOT .STATUS THEN SIGNAL (.STATUS);
; 4184          4289  5                             END;
; 4185          4290  5                         TES;
; 4186          4291  4                     END;
; 4187          4292  3                 END;
; 4188          4293  3
; 4189          4294  3             [DSC$K_DTYPE_ZI]:
; 4190          4295  4                 BEGIN
; 4191          4296  4                 OWN
; 4192          4297  4                     INPUT_STR:  VECTOR[100, BYTE],
; 4193          4298  4                     OUTPUT_STR:  VECTOR[100, BYTE];
; 4194          4299  4                 INPUT_STR[0] = .SRC_INFO[S_LEN];
; 4195          4300  4                 CH$MOVE (.INPUT_STR[0], .SRC_INFO[S_POINTER], INPUT_STR[1]);
; 4196          4301  4                 STATUS = DBG$INS_ENCODE (INPUT_STR, OUTPUT_STR, .DESTINATION[DSC$A_POINTER]);
```

```
;  4197          4302  4                    IF NOT .STATUS THEN SIGNAL (.STATUS);
;  4198          4303  4                    DESTINATION[DSC$W_LENGTH] = .OUTPUT_STR[0];
;  4199          4304  4                    CH$MOVE (.OUTPUT_STR[0], OUTPUT_STR[1], .DESTINATION[DSC$A_POINTER]);
;  4200          4305  3                    END;
;  4201          4306  3
;  4202          4307  3                [OTHERWISE]:
;  4203          4308  3                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  nbds_nbds');
;  4204          4309  3                TES;                                    !For NBDS_NBDS
;  4205          4310  3
;  4206          4311  3            TES;                                        !End of the main CASE statement.
;  4207          4312  3
;  4208          4313  3
;  4209          4314  3        ! If the destination class is unaligned, then the output will be in a temporary buffer.
;  4210          4315  3        ! Copy from the temporary buffer to:  .(destination pointer + number of bits to be offset).
;  4211          4316  3        !
;  4212          4317  3        IF .DESTINATION[DSC$B_CLASS] EQL DSC$K_CLASS_UBS
;  4213          4318  3        THEN
;  4214          4319  4            BEGIN
;  4215          4320  4            SRC_POS = 0;
;  4216          4321  4            DST_POS = .DESTINATION[DSC$L_POS];
;  4217          4322  4            CASE .DESTINATION[DSC$B_DTYPE] FROM K_MIN_DTYPE_STA TO K_MAX_DTYPE_STA OF
;  4218          4323  4                SET
;  4219          4324  4                [DSC$K_DTYPE_V, DSC$K_DTYPE_SV, DSC$K_DTYPE_VU, DSC$K_DTYPE_SVU, DSC$K_DTYPE_TF]:
;  4220          4325  5                    BEGIN
;  4221          4326  5                    MAP
;  4222          4327  5                        DESTINATION_PTR:  REF BITVECTOR,
;  4223          4328  5                        OUTPUT:  REF BITVECTOR;
;  4224          4329  5
;  4225          4330  5                    INCR I FROM 1 TO .DESTINATION[DSC$W_LENGTH] DO
;  4226          4331  6                        BEGIN
;  4227          4332  6                        DESTINATION_PTR[.DST_POS] = .OUTPUT[.SRC_POS];
;  4228          4333  6                        DST_POS = .DST_POS + 1;
;  4229          4334  6                        SRC_POS = .SRC_POS + 1;
;  4230          4335  5                        END;
;  4231          4336  4                    END;
;  4232          4337  4
;  4233          4338  4                [INRANGE]:
;  4234          4339  5                    BEGIN
;  4235          4340  5                    INCR I FROM 1 TO .DESTINATION[DSC$W_LENGTH] DO
;  4236          4341  6                        BEGIN
;  4237          4342  6                        (.DESTINATION_PTR)<.DST_POS, 8> = .(.OUTPUT)<.SRC_POS, 8>;
;  4238          4343  6                        DST_POS = .DST_POS + 8;
;  4239          4344  6                        SRC_POS = .SRC_POS + 8;
;  4240          4345  5                        END;
;  4241          4346  4                    END;
;  4242          4347  4
;  4243          4348  4                [OUTRANGE]:
;  4244          4349  4                    $DBG_ERROR ('DBGCVTDX\DBG$CVT_DX_DX:  invalid dtype');
;  4245          4350  4                TES;
;  4246          4351  3            END;
;  4247          4352  3
;  4248          4353  2        END;                                            ! End the ELSE part of the Absolute Date Time IF
;  4249          4354  2
;  4250          4355  2        ! If output string length is requested then supply it.
;  4251          4356  2        !
;  4252          4357  2        IF ACTUALCOUNT() GTR 2 THEN (.OUTLEN)<0, 16, 0> = .OUTPUT_STR_LEN;
;  4253          4358  2
```

```
; 4254          4359  1     END;                                    ! End of routine DBG$CVT_DX_DX.


                                                                      .PSECT  DBG$PLIT,NOWRT,  SHR,  PIC,0

                                                              00AD2           .BLKB   2
                                  00  00  5C  29  67  49  29  04  00AD4 P.ADX:  .ASCII  <4>\)Ig)\<92><0><0>
                                                  0000FF00  FFFF507F  00ADC P.ADY:  .LONG   ^XFFFF507F, ^X0000FF00
                          00000000  00000000  0000FFFE  FFFF4020  00AE4 P.ADZ:  .LONG   -
                                                                              ^XFFFF4020, ^X0000FFFE, ^X00000000, ^X0000-
                                                                              0000
                                                      00  00  00  0C  00AF4 P.AEA:  .ASCII  <12><0><0><0>
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  34  00AF8 P.AEB:  .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  invalid \
61  76  6E  69  20  20  3A  58  44  5F  58  44  5F  54  56  00B07
                                                  20  64  69  6C  00B16
69  72  63  73  65  64  20  6E  69  20  65  70  79  74  64  00B1A          .ASCII  \dtype in descriptor\
                                                  72  6F  74  70  00B29
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  34  00B2D P.AEC:  .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  invalid \
61  76  6E  69  20  20  3A  58  44  5F  58  44  5F  54  56  00B3C
                                                  20  64  69  6C  00B4B
69  72  63  73  65  64  20  6E  69  20  73  73  61  6C  63  00B4F          .ASCII  \class in descriptor\
                                                  72  6F  74  70  00B5E
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  38  00B62 P.AED:  .ASCII  \8DBGCVTDX\<92>\DBG$CVT_DX_DX:  invalid \
61  76  6E  69  20  20  3A  58  44  5F  58  44  5F  54  56  00B71
                                                  20  64  69  6C  00B80
6D  6F  63  20  65  70  79  74  64  2D  73  73  61  6C  63  00B84          .ASCII  \class-dtype combination\
                                  6E  6F  69  74  61  6E  69  62  00B93
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  39  00B98 P.AEE:  .ASCII  \9DBGCVTDX\<92>\DBG$CVT_DX_DX:  invalid \
61  76  6E  69  20  20  3A  58  44  5F  58  44  5F  54  56  00BAA
                                                  20  64  69  6C  00BB9
74  73  20  65  74  79  62  20  63  69  72  65  6D  75  6E  00BBD          .ASCII  \numeric byte string data\
                                  61  74  61  64  20  67  6E  69  72  00BCC
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  26  00BD5 P.AEF:  .ASCII  \&DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_s\
69  6C  6D  73  20  20  3A  58  44  5F  58  44  5F  54  56  00BE4
                                                  73  5F  74  6E  00BF3
                                                  74  6E  69  6C  6D  00BF7          .ASCII  \mlint\
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  26  00BFC P.AEG:  .ASCII  \&DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_l\
69  6C  6D  73  20  20  3A  58  44  5F  58  44  5F  54  56  00C0B
                                                  6C  5F  74  6E  00C1A
                                                  74  6E  69  67  72  00C1E          .ASCII  \rgint\
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  26  00C23 P.AEH:  .ASCII  \&DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_l\
69  67  72  6C  20  20  3A  58  44  5F  58  44  5F  54  56  00C32
                                                  6C  5F  74  6E  00C41
                                                  74  6E  69  67  72  00C45          .ASCII  \rgint\
                                                                  00C4A           .BLKB   2
                          00000000  00004100                00C4C P.AEI:  .LONG   ^X00004100, ^X00000000
                          00000000  00004100                00C54 P.AEJ:  .LONG   ^X00004100, ^X00000000
                          00000000  00004220                00C5C P.AEK:  .LONG   ^X00004220, ^X00000000
                          00000000  00004220                00C64 P.AEL:  .LONG   ^X00004220, ^X00000000
                          00000000  00004100                00C6C P.AEM:  .LONG   ^X00004100, ^X00000000
                          00000000  00004100                00C74 P.AEN:  .LONG   ^X00004100, ^X00000000
                          00000000  00004220                00C7C P.AEO:  .LONG   ^X00004220, ^X00000000
                          00000000  00004220                00C84 P.AEP:  .LONG   ^X00004220, ^X00000000
                          00000000  00004100                00C8C P.AEQ:  .LONG   ^X00004100, ^X00000000
                          00000000  00004100                00C94 P.AER:  .LONG   ^X00004100, ^X00000000
                          00000000  00004100                00C9C P.AES:  .LONG   ^X00004100, ^X00000000
```

```
                                                00000000  00004100  00CA4 P.AET:  .LONG   ^X00004100, ^X00000000
                                                00000000  00004220  00CAC P.AEU:  .LONG   ^X00004220, ^X00000000
                                                00000000  00004220  00CB4 P.AEV:  .LONG   ^X00004220, ^X00000000
                                                00000000  00004220  00CBC P.AEW:  .LONG   ^X00004220, ^X00000000
                                                00000000  00004220  00CC4 P.AEX:  .LONG   ^X00004220, ^X00000000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B    00CCC P.AEY:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_s\
69 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56    00CDB
                              73 5F 74 6E       00CEA
                  78 6C 70 6D 63 74 6C 66 6C 6D 00CEE         .ASCII  \mlfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B    00CF8 P.AEZ:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_s\
69 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56    00D07
                              73 5F 74 6E       00D16
                  78 6C 70 6D 63 74 6C 66 6C 6D 00D1A         .ASCII  \mlfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 30    00D24 P.AFA:  .ASCII  \0DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\
66 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56    00D33
                              6D 63 74 6C       00D42
78 6C 70 6D 63 74 6C 66 6C 6D 73 5F 78 6C 70    00D46         .ASCII  \plx_smlfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 28    00D55 P.AFB:  .ASCII  \(DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_smlf\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56    00D64
                              66 6C 6D 73       00D73
                  78 6C 70 6D 63 74 6C 77       00D77         .ASCII  \ltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 29    00D7E P.AFC:  .ASCII  \)DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_sml\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56    00D8D
                              6C 6D 73 5F       00D9C
                  78 6C 70 6D 63 74 6C 66       00DA0         .ASCII  \fltcmplx\
                  00000000  00000000  00000000  00004002  00DA8 P.AFD:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00DB8 P.AFE:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  40004004  00DC8 P.AFF:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  40004004  00DD8 P.AFG:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00DE8 P.AFH:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00DF8 P.AFI:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  40004004  00E08 P.AFJ:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  40004004  00E18 P.AFK:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00E28 P.AFL:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00E38 P.AFM:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                  00000000  00000000  00000000  00004002  00E48 P.AFN:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
```

```
00000000  00000000  00000000  00004002  00E58 P.AFO:  .LONG  0000
                                                            -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00E68 P.AFP:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00E78 P.AFQ:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00E88 P.AFR:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00E98 P.AFS:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00EA8 P.AFT:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00EB8 P.AFU:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00EC8 P.AFV:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00ED8 P.AFW:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00EE8 P.AFX:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00EF8 P.AFY:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00F08 P.AFZ:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00F18 P.AGA:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00F28 P.AGB:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00F38 P.AGC:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00F48 P.AGD:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  00004002  00F58 P.AGE:  .LONG  -
                                                            ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00F68 P.AGF:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                            0000
00000000  00000000  00000000  40004004  00F78 P.AGG:  .LONG  -
                                                            ^X40004004, ^X00000000, ^X00000000, ^X0000-
```

```
                        00000000  00000000  00000000  40004004  00F88 P.AGH:  .LONG   0000
                                                                                    -
                                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                                    0000
                        00000000  00000000  00000000  40004004  00F98 P.AGI:  .LONG   0000
                                                                                    -
                                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                                    0000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B 00FA8 P.AGJ:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_l\
69 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 00FB7
                                  6C 5F 74 6E 00FC6
                  78 6C 70 6D 63 74 6C 66 67 72 00FCA          .ASCII  \rgfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B 00FD4 P.AGK:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_l\
69 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 00FE3
                                  6C 5F 74 6E 00FF2
                  78 6C 70 6D 63 74 6C 66 67 72 00FF6          .ASCII  \rgfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 30 01000 P.AGL:  .ASCII  \0DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\
66 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 0100F
                                  6D 63 74 6C 0101E
78 6C 70 6D 63 74 6C 66 67 72 6C 5F 78 6C 70 01022          .ASCII  \plx_lrgfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 30 01031 P.AGM:  .ASCII  \0DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
66 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 01040
                                  6D 63 74 6C 0104F
78 6C 70 6D 63 74 6C 66 67 72 6C 5F 78 6C 70 01053          .ASCII  \plx_lrgfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 28 01062 P.AGN:  .ASCII  \(DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_lrgf\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56 01071
                                  66 67 72 6C 01080
                        78 6C 70 6D 63 74 6C 01084          .ASCII  \ltcmplx\
                                                    01088          .BLKB   1
                                  00 00 00 2C 0108C P.AGO:  .ASCII  \,\<0><0><0>
                                  00 00 00 2C 01090 P.AGP:  .ASCII  \,\<0><0><0>
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 23 01094 P.AGQ:  .ASCII  \#DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_d\
69 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 010A3
                                  64 5F 74 6E 010B2
                                  63 65 010B6          .ASCII  \ec\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 20 010B8 P.AGR:  .ASCII  \ DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_dec\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56 010C7
                                  63 65 64 010D6
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 26 010D9 P.AGS:  .ASCII  \&DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_s\
69 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 010E8
                                  73 5F 74 6E 010F7
                                  74 6E 69 6C 6D 010FB          .ASCII  \mlint\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 23 01100 P.AGT:  .ASCII  \#DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_d\
69 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 0110F
                                  64 5F 74 6E 0111E
                                  63 65 01122          .ASCII  \ec\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 28 01124 P.AGU:  .ASCII  \(DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\
66 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 01133
                                  6D 63 74 6C 01142
                        63 65 64 5F 78 6C 70 01146          .ASCII  \plx_dec\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 28 0114D P.AGV:  .ASCII  \(DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
66 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 0115C
                                  6D 63 74 6C 0116B
                        63 65 64 5F 78 6C 70 0116F          .ASCII  \plx_dec\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 21 01176 P.AGW:  .ASCII  \!DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_dec\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56 01185
                                  63 65 64 5F 01194
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 31 01198 P.AGX:  .ASCII  \1DBGCVTDX\<92>\DBG$CVT_DX_DX inconsiste\
```

```
73 6E 6F 63 6E 69 20 58 44 5F 58 44 5F 54 56 011A7
                                    65 74 73 69 011B6
72 6F 74 63 61 66 20 65 6C 61 63 73 20 74 6E 011BA          .ASCII  \nt scale factors\
                                             73 011C9
                                                011CA          .BLKB   2
                         00000000 00004100 011CC P.AGY:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 011D4 P.AGZ:  .LONG   ^X00004100, ^X00000000
       00000000 00000000 00000000 00004002 011DC P.AHA:  .LONG   -
                                                                 ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                 0000
       00000000 00000000 00000000 00004002 011EC P.AHB:  .LONG   -
                                                                 ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                 0000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 3D 011FC P.AHC:  .ASCII  \=DBGCVTDX\<92>\DBG$CVT_DX_DX scale fact\
20 65 6C 61 63 73 20 58 44 5F 58 44 5F 54 56 0120B
                                    74 63 61 66 0121A
20 64 72 6F 77 61 74 63 6F 20 6E 6F 20 72 6F 0121E          .ASCII  \or on octaword not supported\
   64 65 74 72 6F 70 70 75 73 20 74 6F 6E 0122D
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 42 0123A P.AHD:  .ASCII  \BDBGCVTDX\<92>\DBG$CVT_DX_DX binary sca\
79 72 61 6E 69 62 20 58 44 5F 58 44 5F 54 56 01249
                                    61 63 73 20 0125B
61 70 20 6E 6F 20 72 6F 74 63 61 66 20 65 6C 0125C          .ASCII  \le factor on packed not supported\
72 6F 70 70 75 73 20 74 6F 6E 20 64 65 6B 63 0126B
                                       64 65 74 0127A
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 34 0127D P.AHE:  .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  error in\
6F 72 72 65 20 20 3A 58 44 5F 58 44 5F 54 56 0128C
                                    6E 69 20 72 0129B
72 65 76 6E 6F 63 20 65 74 2D 6F 74 2D 68 20 0129F          .ASCII  \ h-to-te conversion\
                                    6E 6F 69 73 012AE
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 24 012B2 P.AHF:  .ASCII  \$DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlint_n\
69 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 012C1
                                 6E 5F 74 6E 012D0
                                       73 64 62 012D4          .ASCII  \bds\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 24 012D7 P.AHG:  .ASCII  \$DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgint_n\
69 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 012E6
                                 6E 5F 74 6E 012F5
                                       73 64 62 012F9          .ASCII  \bds\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 21 012FC P.AHH:  .ASCII  \!DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_nbds\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56 0130B
                                    73 64 62 6E 0131A
                                                0131E          .BLKB   2
                         00000000 00004100 01320 P.AHI:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 01328 P.AHJ:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 01330 P.AHK:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 01338 P.AHL:  .LONG   ^X00004100, ^X00000000
                         00000000 00004220 01340 P.AHM:  .LONG   ^X00004220, ^X00000000
                         00000000 00004220 01348 P.AHN:  .LONG   ^X00004220, ^X00000000
                         00000000 00004220 01350 P.AHO:  .LONG   ^X00004220, ^X00000000
                         00000000 00004220 01358 P.AHP:  .LONG   ^X00004220, ^X00000000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B 01360 P.AHQ:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\
66 6C 6D 73 20 20 3A 58 44 5F 58 44 5F 54 56 0136F
                                    6D 63 74 6C 0137E
          74 6E 69 6C 6D 73 5F 78 6C 70 01382          .ASCII  \plx_smlint\
                         00000000 00004100 0138C P.AHR:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 01394 P.AHS:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 0139C P.AHT:  .LONG   ^X00004100, ^X00000000
                         00000000 00004100 013A4 P.AHU:  .LONG   ^X00004100, ^X00000000
```

```
                                                00000000  00004220  013AC P.AHV:  .LONG   ^X00004220, ^X00000000                          ;
                                                00000000  00004220  013B4 P.AHW:  .LONG   ^X00004220, ^X00000000                          ;
                                                00000000  00004220  013BC P.AHX:  .LONG   ^X00004220, ^X00000000                          ;
                                                00000000  00004220  013C4 P.AHY:  .LONG   ^X00004220, ^X00000000                          ;
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  2B  013CC P.AHZ:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\              ;
66  6C  6D  73  20  20  3A  58  44  5F  58  44  5F  54  56  013DB
                                                    6D  63  74  6C  013EA
                    74  6E  69  67  72  6C  5F  7B  6C  70  013EE          .ASCII  \plx_lrgint\
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  34  013F8 P.AIA:  .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  error in\           ;
6F  72  72  65  20  20  3A  58  44  5F  58  44  5F  54  56  01407
                                    6E  69  20  72  01416
72  65  76  6E  6F  63  20  65  74  2D  6F  74  2D  64  20  0141A          .ASCII  \ d-to-te conversion\
                                    6E  6F  69  73  01429
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  29  0142D P.AIB:  .ASCII  \)DBGCVTDX\<92>\DBG$CVT_DX_DX:  smlfltcm\              ;
66  6C  6D  73  20  20  3A  58  44  5F  58  44  5F  54  56  0143C
                                    6D  63  74  6C  0144B
                        73  64  62  6E  5F  7B  6C  70  0144F          .ASCII  \plx_nbds\                                              ;
                                                        01457          .BLKB   1
                00000000  00000000  00000000  00004002  01458 P.AIC:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  01468 P.AID:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  01478 P.AIE:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  01488 P.AIF:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  40004004  01498 P.AIG:  .LONG   -                                                       ;
                                                                               ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  40004004  014A8 P.AIH:  .LONG   -                                                       ;
                                                                               ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  40004004  014B8 P.AII:  .LONG   -                                                       ;
                                                                               ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  40004004  014C8 P.AIJ:  .LONG   -                                                       ;
                                                                               ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  014D8 P.AIK:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  014E8 P.AIL:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  014F8 P.AIM:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  00004002  01508 P.AIN:  .LONG   -                                                       ;
                                                                               ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
                00000000  00000000  00000000  40004004  01518 P.AIO:  .LONG   -                                                       ;
                                                                               ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                               0000
```

```
                00000000  00000000  00000000  40004004  01528 P.AIP:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  01538 P.AIQ:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  01548 P.AIR:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
43  24  47  42  44  5C  58  44  54  56  43  47  42  44  2B  01558 P.AIS:  .ASCII  \+DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
66  67  72  6C  20  20  3A  58  44  5F  58  44  5F  54  56  01567
                                                6D  63  74  6C  01576
                74  6E  69  6C  6D  73  5F  78  6C  70  0157A         .ASCII  \plx_smlint\
                00000000  00000000  00000000  00004002  01584 P.AIT:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  01594 P.AIU:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  015A4 P.AIV:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  015B4 P.AIW:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  015C4 P.AIX:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  015D4 P.AIY:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  015E4 P.AIZ:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  015F4 P.AJA:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  01604 P.AJB:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  01614 P.AJC:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  01624 P.AJD:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  00004002  01634 P.AJE:  .LONG  -
                                                                    ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  01644 P.AJF:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  01654 P.AJG:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                    0000
                00000000  00000000  00000000  40004004  01664 P.AJH:  .LONG  -
                                                                    ^X40004004, ^X00000000, ^X00000000, ^X0000-
```

```
                00000000  00000000  00000000  40004004  01674 P.AJI:  .LONG   0000
                                                                              -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 2B 01684 P.AJJ:  .ASCII  \$DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
66 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 01693
                                        6D 63 74 6C 016A2
                74 6E 69 67 72 6C 5F 78 6C 70 016A6        .ASCII  \plx_lrgint\
                00000000  00000000  00000000  00004002  016B0 P.AJK:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  016C0 P.AJL:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  016D0 P.AJM:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  016E0 P.AJN:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  016F0 P.AJO:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01700 P.AJP:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01710 P.AJQ:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01720 P.AJR:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  01730 P.AJS:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  01740 P.AJT:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  01750 P.AJU:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  00004002  01760 P.AJV:  .LONG   -
                                                                              ^X00004002, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01770 P.AJW:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01780 P.AJX:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  01790 P.AJY:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
                00000000  00000000  00000000  40004004  017A0 P.AJZ:  .LONG   -
                                                                              ^X40004004, ^X00000000, ^X00000000, ^X0000-
                                                                              0000
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 30 017B0 P.AKA:  .ASCII  \0DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
```

```
66 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 017BF
                                 6D 63 74 6C 017CE
78 6C 70 6D 63 74 6C 66 6C 6D 73 5F 78 6C 70 017D2          .ASCII  \plx_smlfltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 34 017E1 P.AKB:   .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  error in\
6F 72 72 65 20 20 3A 58 44 5F 58 44 5F 54 56 017F0
                                 6E 69 20 72 017FF
72 65 76 6E 6F 63 20 65 74 2D 6F 74 2D 67 20 01803          .ASCII  \ g-to-te conversion\
                                 6E 6F 69 73 01812
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 34 01816 P.AKC:   .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  error in\
6F 72 72 65 20 20 3A 58 44 5F 58 44 5F 54 56 01825
                                 6E 69 20 72 01834
72 65 76 6E 6F 63 20 65 74 2D 6F 74 2D 68 20 01838          .ASCII  \ h-to-te conversion\
                                 6E 6F 69 73 01847
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 29 0184B P.AKD:   .ASCII  \)DBGCVTDX\<92>\DBG$CVT_DX_DX:  lrgfltcm\
66 67 72 6C 20 20 3A 58 44 5F 58 44 5F 54 56 0185A
                                 6D 63 74 6C 01869
                     73 64 62 6E 5F 78 6C 70 0186D          .ASCII  \plx_nbds\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 23 01875 P.AKE:   .ASCII  \#DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_smli\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56 01884
                                 69 6C 6D 73 01893
                                       74 6E 01897          .ASCII  \nt\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 23 01899 P.AKF:   .ASCII  \#DBGCVTDX\<92>\DBG$CVT_DX_DX:  dec_lrgi\
5F 63 65 64 20 20 3A 58 44 5F 58 44 5F 54 56 018A8
                                 69 67 72 6C 018B7
                                       74 6E 018BB          .ASCII  \nt\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 24 018BD P.AKG:   .ASCII  \$DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_sml\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56 018CC
                                 6C 6D 73 5F 018DB
                                    74 6E 69 018DF          .ASCII  \int\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 24 018E2 P.AKH:   .ASCII  \$DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_lrg\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56 018F1
                                 67 72 6C 5F 01900
                                    74 6E 69 01904          .ASCII  \int\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 29 01907 P.AKI:   .ASCII  \)DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_lrg\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56 01916
                                 67 72 6C 5F 01925
                     78 6C 70 6D 63 74 6C 66 01929          .ASCII  \fltcmplx\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 34 01931 P.AKJ:   .ASCII  \4DBGCVTDX\<92>\DBG$CVT_DX_DX:  error in\
6F 72 72 65 20 20 3A 58 44 5F 58 44 5F 54 56 01940
                                 6E 69 20 72 0194F
72 65 76 6E 6F 63 20 65 74 2D 6F 74 2D 68 20 01953          .ASCII  \ h-to-te conversion\
                                 6E 6F 69 73 01962
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 22 01966 P.AKK:   .ASCII  \"DBGCVTDX\<92>\DBG$CVT_DX_DX:  nbds_nbd\
73 64 62 6E 20 20 3A 58 44 5F 58 44 5F 54 56 01975
                                 64 62 6E 5F 01984
                                          73 01988          .ASCII  \s\
43 24 47 42 44 5C 58 44 54 56 43 47 42 44 26 01989 P.AKL:   .ASCII  \&DBGCVTDX\<92>\DBG$CVT_DX_DX:  invalid \
61 76 6E 69 20 20 3A 58 44 5F 58 44 5F 54 56 01998
                                 20 64 69 6C 019A7
                              65 70 79 74 64 019AB          .ASCII  \dtype\

                                                            .PSECT  DBG$OWN,NOEXE,  PIC,2

                                           00008 INPUT_STR:
                                                            .BLKB   100
                                           0006C OUTPUT_STR:
                                                            .BLKB   100
```

```
                                                        .PSECT  DBG$CODE,NOWRT,  SHR,  PIC,0

                                        0FFC 00000      .ENTRY  DBG$CVT_DX_DX, Save R2,R3,R4,R5,R6,R7,R8,-      ; 1763
                                                                R9,R10,R11
                                5E      FEE8   CE   9E 00002    MOVAB   -280(SP), SP
                                6D      3091   CF   DE 00007    MOVAL   661$, (FP)                              ; 1930
                                59        08   AC   D0 0000C    MOVL    DESTINATION, R9                         ; 1993
                                5B        02   A9   9E 00010    MOVAB   2(R9), R11
                                               51   D4 00014    CLRL    R1
                                23             6B   91 00016    CMPB    (R11), #35
                                               04   12 00019    BNEQ    1$
                                               51   D6 0001B    INCL    R1
                                               0D   11 0001D    BRB     2$
                                50        04   AC   D0 0001F 1$: MOVL    SOURCE, R0                          ; 1994
                                23        02   A0   91 00023    CMPB    2(R0), #35
                                               03   13 00027    BEQL    2$
                                             0091   31 00029    BRW     8$
                                0E             6B   91 0002C 2$: CMPB    (R11), #14                          ; 1996
                                               44   12 0002F    BNEQ    4$
                                50        04   AC   D0 00031    MOVL    SOURCE, R0                          ; 1997
                                23        02   A0   91 00035    CMPB    2(R0), #35
                                               3A   12 00039    BNEQ    4$
                        58      AE 010E0017    8F   D0 0003B    MOVL    #17694743, CLASS_S_DESC             ; 2004
                        5C      AE        04   A9   D0 00043    MOVL    4(R9), CLASS_S_DESC+4               ; 2005
                                               7E   D4 00048    CLRL    -(SP)                               ; 2006
                                50        04   AC   D0 0004A    MOVL    SOURCE, R0
                                               04   A0   DD 0004E    PUSHL   4(R0)
                                               60   AE   9F 00051    PUSHAB  CLASS_S_DESC
                                               30   AE   9F 00054    PUSHAB  TEMP
                00000000G       00                  04   FB 00057    CALLS   #4, SYS$ASCTIM
                                0D                 50   E8 0005E    BLBS    R0, 3$
                                        000282A0   8F   DD 00061    PUSHL   #164512                             ; 2008
                00000000G       00                  01   FB 00067    CALLS   #1, LIB$SIGNAL
                             14 AE           24   AE   D0 0006E 3$: MOVL    TEMP, OUTPUT_STR_LEN                ; 2009
                                               45   11 00073    BRB     7$                                  ; 1996
                                35                 51   E9 00075 4$: BLBC    R1, 5$                              ; 2012
                                50        04   AC   D0 00078    MOVL    SOURCE, R0                          ; 2013
                                0E        02   A0   91 0007C    CMPB    2(R0), #14
                                               2B   12 00080    BNEQ    5$
                             5A AE        010E   8F   B0 00082    MOVW    #270, CLASS_S_DESC+2                ; 2017
                                50        04   AC   D0 00088    MOVL    SOURCE, R0                          ; 2018
                             58 AE               60   B0 0008C    MOVW    (R0), CLASS_S_DESC
                             5C AE        04   A0   D0 00090    MOVL    4(R0), CLASS_S_DESC+4               ; 2019
                                               04   A9   DD 00095    PUSHL   4(R9)                               ; 2020
                                               5C   AE   9F 00098    PUSHAB  CLASS_S_DESC
                00000000G       00                  02   FB 0009B    CALLS   #2, SYS$BINTIM
                                15                 50   E8 000A2    BLBS    R0, 7$
                                        00028F88   8F   DD 000A5    PUSHL   #167816                             ; 2022
                                               06   11 000AB    BRB     6$
                                        000287D8   8F   DD 000AD 5$: PUSHL   #165848                            ; 2025
                00000000G       00                  01   FB 000B3 6$: CALLS   #1, LIB$SIGNAL
                                             2FD4   31 000BA 7$: BRW     659$                                ; 1996
                                03             6C   91 000BD 8$: CMPB    (AP), #3                            ; 2030
                                               07   1B 000C0    BLEQU   9$
```

```
                                0C   AE      10   AC  D0 000C2          MOVL    16(AP), CVT_ROUND_FLAG                  ; 2032
                                                  03  11 000C7          BRB     10$
                                0C   AE          0C  AE  D4 000C9 9$:    CLRL    CVT_ROUND_FLAG                          ; 2034
                                     50          04  AC  D0 000CC 10$:   MOVL    SOURCE, R0                              ; 2039
                                     15          02  A0  91 000D0        CMPB    2(R0), #21
                                                     12  12 000D4        BNEQ    11$
                                     0E              6B  91 000D6        CMPB    (R11), #14                              ; 2040
                                                     0D  13 000D9        BEQL    11$
                                                     50  DD 000DB        PUSHL   R0                                      ; 2042
                            00000000G   00           01  FB 000DD        CALLS   #1, DBG$STRIP_ZEROES
                                    04  AC           50  D0 000E4        MOVL    R0, SOURCE
                                        5A      04  AC  D0 000E8 11$:    MOVL    SOURCE, R10                             ; 2052
                                    04  AE      02  AA  9E 000EC        MOVAB   2(R10), 4(SP)
                                                     51  D4 000F1        CLRL    R1
                                     0F      04  BE  91 000F3          CMPB    @4(SP), #15
                                                     02  1F 000F7        BLSSU   12$
                                                     51  D6 000F9        INCL    R1
                                                     50  D4 000FB 12$:   CLRL    R0                                      ; 2053
                                     15      04  BE  91 000FD          CMPB    @4(SP), #21
                                                     02  1A 00101        BGTRU   13$
                                                     50  D6 00103        INCL    R0
                                     52              51  D2 00105 13$:   MCOML   R1, R2
                        00000000'    EF              52  CB 00108        BICL3   R2, R0, DECIMAL_CONVERT
                                     0D      03  A9  91 00110          CMPB    3(R9), #13                              ; 2061
                                                     0C  12 00114        BNEQ    14$
                                34   AE      D0  AD  9E 00116          MOVAB   OUTPUT_BUFFER, OUTPUT                   ; 2064
                                20   AE      04  A9  D0 0011B          MOVL    4(R9), DESTINATION_PTR                  ; 2065
                                                     05  11 00120        BRB     15$                                     ; 2061
                                34   AE      04  A9  D0 00122 14$:    MOVL    4(R9), OUTPUT                           ; 2068
                08          00              6E      00  2C 00127 15$:   MOVC5   #0, (SP), #0, #8, SRC_INFO              ; 2073
                                                F8  AD      0012C
                08          00              6E      00  2C 0012E        MOVC5   #0, (SP), #0, #8, DST_INFO              ; 2074
                                                F0  AD      00133
                20          00              6E      00  2C 00135        MOVC5   #0, (SP), #0, #32, INTMED_DATA          ; 2075
                                                B0  AD      0013A
                32          20              6E      00  2C 0013C        MOVC5   #0, (SP), #32, #50, TEMP_BUF1           ; 2076
                                              FF7C  CD      00141
                32          20              6E      00  2C 00144        MOVC5   #0, (SP), #32, #50, TEMP_BUF2           ; 2077
                                                60  AE      00149
                                                14  AE  D4 0014B        CLRL    OUTPUT_STR_LEN                          ; 2078
                                        5A  AE      010E  8F  B0 0014E  MOVW    #270, CLASS_S_DESC+2                    ; 2085
                                        1C  AE  00000000' EF  9E 00154  MOVAB   P.ADX, LRGST_P_LU                       ; 2091
                                        10  AE  00000000' EF  9E 0015C  MOVAB   P.ADY, LRGST_D_LU                       ; 2092
                                        18  AE  00000000' EF  9E 00164  MOVAB   P.ADZ, LRGST_H_LU                       ; 2093
                                        08  AE  00000000' EF  9E 0016C  MOVAB   P.AEA, PACK_ZERO                        ; 2094
                                        F9  AD          B0  AD  9E 00174  MOVAB   INTMED_DATA, SRC_INFO+1               ; 2102
                                        FD  AD              20  B0 00179  MOVW    #32, SRC_INFO+5                       ; 2103
                                                28  AE      9F 0017D    PUSHAB  CVT_PATH                                ; 2110
                                                F0  AD      9F 00180    PUSHAB  DST_INFO
                                                F8  AD      9F 00183    PUSHAB  SRC_INFO
                                                    59      DD 00186    PUSHL   R9
                                                    5A      DD 00188    PUSHL   R10
                                            0000V   CF          05  FB 0018A  CALLS   #5, FIND_CVT_PATH
                                                    6E          50  D0 0018F  MOVL    R0, STATUS
                                                                43  18 00192  BGEQ    22$                             ; 2119
                                          06 FFFFFFF9   8F          6E  CF 00194  CASEL   STATUS, #-7, #6                 ; 2122
                001E                0016            000E        0026      0019C 16$:  .WORD   20$-16$,-
```

```
                0016            000E            001E    001A4                   17$-16$,-
                                                                                18$-16$,-
                                                                                19$-16$,-
                                                                                19$-16$,-
                                                                                17$-16$,-
                                                                                18$-16$
                                00000000'   EF  9F  001AA 17$:    PUSHAB  P.AEB                           2124
                                            16  11  001B0         BRB     21$
                                00000000'   EF  9F  001B2 18$:    PUSHAB  P.AEC                           2125
                                            0E  11  001B8         BRB     21$
                                00000000'   EF  9F  001BA 19$:    PUSHAB  P.AED                           2126
                                            06  11  001C0         BRB     21$
                                00000000'   EF  9F  001C2 20$:    PUSHAB  P.AEE                           2127
                                            01  DD  001C8 21$:    PUSHL   #1
                                00028362    8F  DD  001CA         PUSHL   #164706
                    00000000G   00          03  FB  001D0         CALLS   #3, LIB$SIGNAL
                                    00E0    8F  B8  001D7 22$:    BISPSW  #224                            2136
                04      FF  AD              01  E1  001DB         BBC     #1, SRC_INFO+7, 23$             2137
                                            51  D4  001E0         CLRL    R1
                                            04  11  001E2         BRB     24$
                            51      F8  AD  98  001E4 23$:    CVTBL   SRC_INFO, R1
                04      F7  AD              01  E1  001E8 24$:    BBC     #1, DST_INFO+7, 25$             2138
                                            50  D4  001ED         CLRL    R0
                                            04  11  001EF         BRB     26$
                            50      F0  AD  98  001F1 25$:    CVTBL   DST_INFO, R0
            30  AE          51              50  C3  001F5 26$:    SUBL3   R0, R1, SCALE
                06      FF  AD              01  E1  001FA         BBC     #1, SRC_INFO+7, 27$             2139
                            51      F8  AD  98  001FF         CVTBL   SRC_INFO, R1
                                            02  11  00203         BRB     28$
                                            51  D4  00205 27$:    CLRL    R1
                    06      F7  AD              01  E1  00207 28$:    BBC     #1, DST_INFO+7, 29$             2140
                                50      F0  AD  98  0020C         CVTBL   DST_INFO, R0
                                            02  11  00210         BRB     30$
                                            50  D4  00212 29$:    CLRL    R0
                    58              51          50  C3  00214 30$:    SUBL3   R0, R1, BIN_SCALE
                                56      28  AE  D0  00218         MOVL    CVT_PATH, R6                    2187
                    23              01          56  CF  0021C         CASEL   R6, #1, #35
        0588            02C7            0119        0048    00220 31$:    .WORD   32$-31$,-
        0119            0CA3            118F        09FF    00228         .WORD   46$-31$,-
        118F            0E5B            0588        02C7    00230         .WORD   76$-31$,-
        0588            02C7            165E        1528    00238         .WORD   110$-31$,-
        1A93            1831            17A3        0E5B    00240         .WORD   159$-31$,-
        1E9A            0E5B            0588        1CAB    00248         .WORD   277$-31$,-
        0588            02C7            239B        1FEE    00250         .WORD   198$-31$,-
        2852            251A            118F        09FF    00258         .WORD   46$-31$,-
        2AE4            0E5B            2A33        02C7    00260         .WORD   76$-31$,-
                                                                                110$-31$,-
                                                                                227$-31$,-
                                                                                277$-31$,-
                                                                                332$-31$,-
                                                                                344$-31$,-
                                                                                76$-31$,-
                                                                                110$-31$,-
                                                                                227$-31$,-
                                                                                357$-31$,-
                                                                                368$-31$,-
                                                                                390$-31$,-
```

```
                                                                 410$-31$,-               ;
                                                                 110$-31$,-               ;
                                                                 227$-31$,-               ;
                                                                 430$-31$,-               ;
                                                                 449$-31$,-               ;
                                                                 503$-31$,-               ;
                                                                 76$-31$,-                ;
                                                                 110$-31$,-               ;
                                                                 159$-31$,-               ;
                                                                 277$-31$,-               ;
                                                                 519$-31$,-               ;
                                                                 575$-31$,-               ;
                                                                 76$-31$,-                ;
                                                                 599$-31$,-               ;
                                                                 227$-31$,-               ;
                                                                 607$-31$                 ;
                              58  D5 00268 32$:     TSTL         BIN_SCALE                ; 2191
                              08  15 0026A          BLEQ         33$                      ;
              B0   AD         02  C4 0026C          MULL2        #2, INTMED_DATA          ;
                              58  D7 00270          DECL         BIN_SCALE                ;
                              F4  11 00272          BRB          32$                      ;
                              08  18 00274 33$:     BGEQ         34$                      ;
              B0   AD         02  C6 00276          DIVL2        #2, INTMED_DATA          ;
                              58  D6 0027A          INCL         BIN_SCALE                ;
                              F6  11 0027C          BRB          33$                      ;
                         30   AE  D5 0027E 34$:     TSTL         SCALE                    ;
                              09  15 00281          BLEQ         35$                      ;
              B0   AD         0A  C4 00283          MULL2        #10, INTMED_DATA         ;
                         30   AE  D7 00287          DECL         SCALE                    ;
                              F2  11 0028A          BRB          34$                      ;
                              09  18 0028C 35$:     BGEQ         36$                      ;
              B0   AD         0A  C6 0028E          DIVL2        #10, INTMED_DATA         ;
                         30   AE  D6 00292          INCL         SCALE                    ;
                              F5  11 00295          BRB          35$                      ;
               29        01   6B  8F 00297 36$:     CASEB        (R11), #1, #41           ; 2193
    0054     006E      005D     0082     0029B 37$:     .WORD        42$-37$,-                ;
    0DBD     0D85      0D5B     0054     002A3                       39$-37$,-                ;
    0054     0054      0054     0054     002AB                       40$-37$,-                ;
    0054     0054      0054     0054     002B3                       38$-37$,-                ;
    0054     0054      0054     0054     002BB                       38$-37$,-                ;
    0054     0054      0054     0054     002C3                       214$-37$,-               ;
    0054     0054      0054     0054     002CB                       217$-37$,-               ;
    0054     0054      0054     0054     002D3                       221$-37$,-               ;
    0054     0054      0082     0054     002DB                       38$-37$,-                ;
    0082     0054      0054     0054     002E3                       38$-37$,-                ;
                       0082     0082     002EB                       38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
                                                                 38$-37$,-                ;
```

```
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        42$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        38$-37$,-
                                                        42$-37$,-
                                                        42$-37$,-
                                                        42$-37$
                00000000'  EF   9F 002EF 38$:   PUSHAB  P.AEF                                   ;2230
                       2CD1 31 002F5           BRW     647$
           50       B0  AD  D0 002F8 39$:       MOVL    INTMED_DATA, R0                         ;2198
        34 BE           50  90 002FC           MOVB    R0, @OUTPUT
 000000FF 8F            50  D1 00300           CMPL    R0, #255
                        0F  11 00307           BRB     41$
           50       B0  AD  D0 00309 40$:       MOVL    INTMED_DATA, R0                         ;2204
        34 BE           50  B0 0030D           MOVW    R0, @OUTPUT
 0000FFFF 8F            50  D1 00311           CMPL    R0, #65535
                        1C  1B 00318 41$:       BLEQU   45$
                    26F2    31 0031A           BRW     565$                                    ;2205
               52       F5  AD  3C 0031D 42$:   MOVZWL  DST_INFO+5, R2                          ;2223
               50           01  CE 00321       MNEGL   #1, I                                   ;2225
                            0C  11 00324       BRB     44$
     51        B0  AD          01  EF 00326 43$:   EXTZV   I, #1, INTMED_DATA, R1
  34 BE           01          50  F0 0032C       INSV    R1, I, #1, @OUTPUT
                  F0          50  F2 00332 44$:   AOBLSS  R2, I, 43$                          ;2223
                          2C9F    31 00336 45$:   BRW     649$                                    ;2187
               02               56  D1 00339 46$:   CMPL    R6, #2                              ;2239
                                5D  12 0033C       BNEQ    52$
                       B0  AD  D5 0033E           TSTL    INTMED_DATA                         ;2240
                                11  18 00341       BGEQ    48$
               50       B0  AD  D0 00343           MOVL    INTMED_DATA, R0
                                03  18 00347       BGEQ    47$
               50               50  CE 00349       MNEGL   R0, R0
          B0   AD                  50  D0 0034C 47$:   MOVL    R0, INTMED_DATA
          FF   AD                  01  88 00350       BISB2   #1, SRC_INFO+7
                       30  AE  D5 00354 48$:   TSTL    SCALE
                                0F  15 00357       BLEQ    49$
               50       B0  AD  9E 00359           MOVAB   INTMED_DATA, R0
                 00000000G  00  16 0035D           JSB     LIB$$CVT_SCALE_OU_UP_BY_10_R1
                       30  AE  D7 00363           DECL    SCALE
                                EC  11 00366       BRB     48$
                                0F  18 00368 49$:   BGEQ    50$
               50       B0  AD  9E 0036A           MOVAB   INTMED_DATA, R0
                 00000000G  00  16 0036E           JSB     LIB$$CVT_SCALE_OU_DOWN_BY_10_R1
                       30  AE  D6 00374           INCL    SCALE
```

```
                                EF  11 00377                    BRB     49$                                     ;
                                58  D5 00379 50$:                TSTL    BIN_SCALE                               ;
                                0E  15 0037B                    BLEQ    51$                                     ;
                          50    B0  AD 9E 0037D                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 00381                    JSB     DBG$CVT_SCALE_OU_UP_BY_2_R1             ;
                                58  D7 00387                    DECL    BIN_SCALE                               ;
                                EE  11 00389                    BRB     50$                                     ;
                                5A  18 0038B 51$:               BGEQ    57$                                     ;
                          50    B0  AD 9E 0038D                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 00391                    JSB     DBG$CVT_SCALE_OU_DOWN_BY_2_R1           ;
                                58  D6 00397                    INCL    BIN_SCALE                               ;
                                F0  11 00399                    BRB     51$                                     ;
                          08        56  D1 0039B 52$:           CMPL    R6, #8                                  ; 2244
                                47  12 0039E                    BNEQ    57$                                     ;
                             30   AE  D5 003A0 53$:              TSTL    SCALE                                   ; 2245
                                0F  15 003A3                    BLEQ    54$                                     ;
                          50    B0  AD 9E 003A5                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 003A9                    JSB     LIB$$CVT_SCALE_OU_UP_BY_10_R1           ;
                             30 AE  D7 003AF                    DECL    SCALE                                   ;
                                EC  11 003B2                    BRB     53$                                     ;
                                0F  18 003B4 54$:               BGEQ    55$                                     ;
                          50    B0  AD 9E 003B6                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 003BA                    JSB     LIB$$CVT_SCALE_OU_DOWN_BY_10_R1         ;
                             30 AE  D6 003C0                    INCL    SCALE                                   ;
                                EF  11 003C3                    BRB     54$                                     ;
                                58  D5 003C5 55$:               TSTL    BIN_SCALE                               ;
                                0E  15 003C7                    BLEQ    56$                                     ;
                          50    B0  AD 9E 003C9                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 003CD                    JSB     DBG$CVT_SCALE_OU_UP_BY_2_R1             ;
                                58  D7 003D3                    DECL    BIN_SCALE                               ;
                                EE  11 003D5                    BRB     55$                                     ;
                                0E  18 003D7 56$:               BGEQ    57$                                     ;
                          50    B0  AD 9E 003D9                 MOVAB   INTMED_DATA, R0                         ;
                    00000000G   00  16 003DD                    JSB     DBG$CVT_SCALE_OU_DOWN_BY_2_R1           ;
                                58  D6 003E3                    INCL    BIN_SCALE                               ;
                                F0  11 003E5                    BRB     56$                                     ;
                          04        68  91 003E7 57$:           CMPB    (R11), #4                               ; 2253
                                31  12 003EA                    BNEQ    59$                                     ;
                50        B4  AD      B8  AD C9 003EC           BISL3   INTMED_DATA+8, INTMED_DATA+4, R0        ; 2255
                              50    BC  AD C8 003F2             BISL2   INTMED_DATA+12, R0                      ;
                                15  13 003F6                    BEQL    58$                                     ;
                          00000000G   00  DD 003F8              PUSHL   DBG$GL_OPCODE_NAME                      ; 2256
                                01  DD 003FE                    PUSHL   #1                                      ;
                          000286A3  8F  DD 00400                PUSHL   #165539                                 ;
                00000000G   00      03  FB 00406               CALLS   #3, LIB$SIGNAL                          ;
                          34    BE      B0  AD D0 00410 58$:    MOVL    INTMED_DATA, @OUTPUT                    ; 2257
                          60    FF      AD  E9 00412            BLBC    SRC_INFO+7, 64$                         ; 2258
                          34    BE      34  BE CE 00416         MNEGL   @OUTPUT, @OUTPUT                        ; 2260
                                59  11 0041B                    BRB     64$                                     ; 2250
                          05        68  91 0041D 59$:           CMPB    (R11), #5                               ; 2263
                                05  13 00420                    BEQL    60$                                     ;
                          09        68  91 00422                CMPB    (R11), #9                               ;
                                51  12 00425                    BNEQ    65$                                     ;
                50        B8  AD      BC  AD C9 00427 60$:      BISL3   INTMED_DATA+12, INTMED_DATA+8, R0       ; 2265
                                15  13 0042D                    BEQL    61$                                     ;
                          00000000G   00  DD 0042F              PUSHL   DBG$GL_OPCODE_NAME                      ; 2266
                                01  DD 00435                    PUSHL   #1                                      ;
```

```
                                000286A3   8F   DD 00437          PUSHL   #165539
                00000000G   00             03   FB 0043D          CALLS   #3, LIB$SIGNAL
                        26      FF      AD   E9 00444 61$:       BLBC    SRC_INFO+7, 63$                                          2267
                                B0      AD   D5 00448           TSTL    INTMED_DATA                                              2269
                                        14   12 0044B           BNEQ    62$
                80000000    BF          B4   AD   D1 0044D      CMPL    INTMED_DATA+4, #-2147483648                              2272
                                        17   13 00455           BEQL    63$
                      B4    AD          B4   AD   D2 00457      MCOML   INTMED_DATA+4, INTMED_DATA+4                           2275
                                        B4   AD   D6 0045C      INCL    INTMED_DATA+4                                            2276
                                        0D   11 0045F           BRB     63$                                                      2269
                      B0    AD          B0   AD   D2 00461 62$: MCOML   INTMED_DATA, INTMED_DATA                               2281
                      B4    AD          B4   AD   D2 00466      MCOML   INTMED_DATA+4, INTMED_DATA+4                           2282
                                        B0   AD   D6 0046B      INCL    INTMED_DATA                                              2283
                            50      34   AE   D0 0046E 63$:     MOVL    OUTPUT, R0                                               2285
                            60      B0   AD   7D 00472          MOVQ    INTMED_DATA, (R0)
                                        51   11 00476 64$:      BRB     72$                                                      2250
                            1A          6B   91 00478 65$:      CMPB    (R11), #26                                               2289
                                        4F   12 0047B           BNEQ    73$
                            42      FF      AD   E9 0047D      BLBC    SRC_INFO+7, 71$                                          2294
                                    B0      AD   D5 00481      TSTL    INTMED_DATA                                              2302
                                            14   12 00484      BNEQ    66$
                                    B4      AD   D5 00486      TSTL    INTMED_DATA+4                                            2303
                                            0F   12 00489      BNEQ    66$
                                    B8      AD   D5 0048B      TSTL    INTMED_DATA+8                                            2304
                                            0A   12 0048E      BNEQ    66$
                80000000    BF          BC   AD   D1 00490      CMPL    INTMED_DATA+12, #-2147483648                             2305
                                        29   13 00498           BEQL    71$
                                        50   D4 0049A 66$:      CLRL    NEXT_LONGWORD                                            2313
                      B0 AD40           B0 AD40   D2 0049C 67$: MCOML   INTMED_DATA[NEXT_LONGWORD], INTMED_DATA-               2315
                                                                [NEXT_LONGWORD]
                F5              50              03   F3 004A3      AOBLEQ  #3, NEXT_LONGWORD, 67$                              2314
                                50              D4   004A7      CLRL    NEXT_LONGWORD                                            2319
                                51      B0 AD40   DE 004A9 68$: MOVAL   INTMED_DATA[NEXT_LONGWORD], R1                       2320
                    FFFFFFFF    BF              61   D1 004AE      CMPL    (R1), #-1
                                                04   12 004B5      BNEQ    69$
                                                61   D4 004B7      CLRL    (R1)                                             2322
                                                04   11 004B9      BRB     70$
                                                61   D6 004BB 69$: INCL    (R1)                                             2326
                                                04   11 004BD      BRB     71$                                              2324
                E6              50              03   F3 004BF 70$: AOBLEQ  #3, NEXT_LONGWORD, 68$                          2320
            34  BE      B0      AD              10   28 004C3 71$: MOVC3   #16, INTMED_DATA, @OUTPUT                        2332
                                            2B0C   31 004C9 72$: BRW     649$                                             2250
                            02                  56   D1 004CC 73$: CMPL    R6, #2                                           2338
                                                08   12 004CF      BNEQ    74$
                                00000000'       EF   9F 004D1      PUSHAB  P.AEG                                            2339
                                                0B   11 004D7      BRB     75$
                            08                  56   D1 004D9 74$: CMPL    R6, #8                                           2341
                                                EB   12 004DC      BNEQ    72$
                                00000000'       EF   9F 004DE      PUSHAB  P.AEH                                            2342
                                            2AE2   31 004E4 75$: BRW     647$
                            03                  56   D1 004E7 76$: CMPL    R6, #3                                           2352
                                                68   12 004EA      BNEQ    81$
                      B0    AD          B0    AD   6E 004EC      CVTLD   INTMED_DATA, INTMED_DATA                            2353
                                                58   D5 004F1 77$: TSTL    BIN_SCALE
                                                15   15 004F3      BLEQ    78$
                            51          B0    AD   9E 004F5      MOVAB   INTMED_DATA, R1
                            50  00000000'   EF   9E 004F9      MOVAB   P.AEI, R0
```

```
                    00000000G  00  16  00500          JSB     DBG$CVT_MULD2_R1                                    ;
                               58  D7  00506          DECL    BIN_SCALE                                           ;
                               E7  11  00508          BRB     77$                                                 ;
                               15  18  0050A  78$:     BGEQ    79$                                                 ;
                            51  B0  AD  9E  0050C          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  00510     MOVAB   P.AEJ, R0                                       ;
                               00000000G  00  16  00517          JSB     DBG$CVT_DIVD2_R1                        ;
                               58  D6  0051D          INCL    BIN_SCALE                                           ;
                               E9  11  0051F          BRB     78$                                                 ;
                            30  AE  D5  00521  79$:     TSTL    SCALE                                               ;
                               16  15  00524          BLEQ    80$                                                 ;
                            51  B0  AD  9E  00526          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  0052A     MOVAB   P.AEK, R0                                       ;
                               00000000G  00  16  00531          JSB     DBG$CVT_MULD2_R1                        ;
                            30  AE  D7  00537          DECL    SCALE                                               ;
                               E5  11  0053A          BRB     79$                                                 ;
                               7C  18  0053C  80$:     BGEQ    85$                                                 ;
                            51  B0  AD  9E  0053E          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  00542     MOVAB   P.AEL, R0                                       ;
                               00000000G  00  16  00549          JSB     DBG$CVT_DIVD2_R1                        ;
                            30  AE  D6  0054F          INCL    SCALE                                               ;
                               E8  11  00552          BRB     80$                                                 ;
                            09  56  D1  00554  81$:     CMPL    R6, #9                                              ; 2357
                               7C  12  00557          BNEQ    87$                                                 ;
                            51  FF7C  CD  9E  00559          MOVAB   TEMP_BUF1, R1                               ; 2358
                            50  B0  AD  9E  0055E          MOVAB   INTMED_DATA, R0                                 ;
                               00000000G  00  16  00562          JSB     DBG$CVT_CVTROUD_R1                      ;
BO  AD  FF7C  CD                  08  28  00568          MOVC3   #8, TEMP_BUF1, INTMED_DATA                      ;
                               58  D5  0056F  82$:     TSTL    BIN_SCALE                                           ;
                               15  15  00571          BLEQ    83$                                                 ;
                            51  B0  AD  9E  00573          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  00577     MOVAB   P.AEM, R0                                       ;
                               00000000G  00  16  0057E          JSB     DBG$CVT_MULD2_R1                        ;
                               58  D7  00584          DECL    BIN_SCALE                                           ;
                               E7  11  00586          BRB     82$                                                 ;
                               15  18  00588  83$:     BGEQ    84$                                                 ;
                            51  B0  AD  9E  0058A          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  0058E     MOVAB   P.AEN, R0                                       ;
                               00000000G  00  16  00595          JSB     DBG$CVT_DIVD2_R1                        ;
                               58  D6  0059B          INCL    BIN_SCALE                                           ;
                               E9  11  0059D          BRB     83$                                                 ;
                            30  AE  D5  0059F  84$:     TSTL    SCALE                                               ;
                               16  15  005A2          BLEQ    85$                                                 ;
                            51  B0  AD  9E  005A4          MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  005A8     MOVAB   P.AEO, R0                                       ;
                               00000000G  00  16  005AF          JSB     DBG$CVT_MULD2_R1                        ;
                            30  AE  D7  005B5          DECL    SCALE                                               ;
                               E5  11  005B8          BRB     84$                                                 ;
                               03  19  005BA  85$:     BLSS    86$                                                 ;
                            0179  31  005BC          BRW     98$                                                 ;
                            51  B0  AD  9E  005BF  86$:     MOVAB   INTMED_DATA, R1                                 ;
                            50 00000000'EF  9E  005C3     MOVAB   P.AEP, R0                                       ;
                               00000000G  00  16  005CA          JSB     DBG$CVT_DIVD2_R1                        ;
                            30  AE  D6  005D0          INCL    SCALE                                               ;
                               E5  11  005D3          BRB     85$                                                 ;
                            0F  56  D1  005D5  87$:     CMPL    R6, #15                                             ; 2362
                               03  13  005D8          BEQL    88$                                                 ;
```

```
                     00AA  31  005DA          BRW     93$                                     ;
                       58  D5  005DD  88$:    TSTL    BIN_SCALE                               ; 2363
                       26  15  005DF          BLEQ    89$                                     ;
                 51     B0  AD  9E  005E1     MOVAB   INTMED_DATA, R1                         ;
                 50 00000000'  EF  9E  005E5 MOVAB   P.AEQ, R0                               ;
                    00000000G  00  16  005EC  JSB     DBG$CVT_MULD2_R1                        ;
                 51     B8  AD  9E  005F2     MOVAB   INTMED_DATA+8, R1                       ;
                 50 00000000'  EF  9E  005F6 MOVAB   P.AER, R0                               ;
                    00000000G  00  16  005FD  JSB     DBG$CVT_MULD2_R1                        ;
                       58  D7  00603          DECL    BIN_SCALE                               ;
                       D6  11  00605          BRB     88$                                     ;
                       26  18  00607  89$:    BGEQ    90$                                     ;
                 51     B0  AD  9E  00609     MOVAB   INTMED_DATA, R1                         ;
                 50 00000000'  EF  9E  0060D MOVAB   P.AES, R0                               ;
                    00000000G  00  16  00614  JSB     DBG$CVT_DIVD2_R1                        ;
                 51     B8  AD  9E  0061A     MOVAB   INTMED_DATA+8, R1                       ;
                 50 00000000'  EF  9E  0061E MOVAB   P.AET, R0                               ;
                    00000000G  00  16  00625  JSB     DBG$CVT_DIVD2_R1                        ;
                       58  D6  0062B          INCL    BIN_SCALE                               ;
                       D8  11  0062D          BRB     89$                                     ;
                       30  AE  D5  0062F  90$:    TSTL    SCALE                               ;
                       27  15  00632          BLEQ    91$                                     ;
                 51     B0  AD  9E  00634     MOVAB   INTMED_DATA, R1                         ;
                 50 00000000'  EF  9E  00638 MOVAB   P.AEU, R0                               ;
                    00000000G  00  16  0063F  JSB     DBG$CVT_MULD2_R1                        ;
                 51     B8  AD  9E  00645     MOVAB   INTMED_DATA+8, R1                       ;
                 50 00000000'  EF  9E  00649 MOVAB   P.AEV, R0                               ;
                    00000000G  00  16  00650  JSB     DBG$CVT_MULD2_R1                        ;
                       30  AE  D7  00656          DECL    SCALE                               ;
                       D4  11  00659          BRB     90$                                     ;
                       03  19  0065B  91$:    BLSS    92$                                     ;
                     00D8  31  0065D          BRW     98$                                     ;
                 51     B0  AD  9E  00660  92$:    MOVAB   INTMED_DATA, R1                     ;
                 50 00000000'  EF  9E  00664 MOVAB   P.AEW, R0                               ;
                    00000000G  00  16  0066B  JSB     DBG$CVT_DIVD2_R1                        ;
                 51     B8  AD  9E  00671     MOVAB   INTMED_DATA+8, R1                       ;
                 50 00000000'  EF  9E  00675 MOVAB   P.AEX, R0                               ;
                    00000000G  00  16  0067C  JSB     DBG$CVT_DIVD2_R1                        ;
                       30  AE  D6  00682          INCL    SCALE                               ;
                       D4  11  00685          BRB     91$                                     ;
                 1B        56  D1  00687  93$:    CMPL    R6, #27                             ; 2367
                           6B  12  0068A          BNEQ    96$                                 ;
                 2C  AE  FD  AD  3C  0068C      MOVZWL  SRC_INFO+5, NO_DIGITS               ; 2368
                 09  03  AA  91  00691          CMPB    3(R10), #9                          ;
                           0A  12  00695          BNEQ    94$                                 ;
                 30  AE  08  AA  98  00697      CVTBL   8(R10), SCALE                       ;
                 30  AE  30  AE  CE  0069C      MNEGL   SCALE, SCALE                        ;
FF7C  CD  2C  AE  B0  AD  2C  AE  08  006A1  94$:    CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -
                                                        TEMP_BUF1
           58  AE  2C  AE        01  A1  006AB      ADDW3   #1, NO_DIGITS, CLASS_S_DESC         ;
                 5C  AE  FF7C  CD  9E  006B1      MOVAB   TEMP_BUF1, CLASS_S_DESC+4           ;
                       7E  44  8F  9A  006B7      MOVZBL  #68, -(SP)                          ;
                           34  AE  DD  006BB      PUSHL   SCALE                               ;
                               7E  D4  006BE      CLRL    -(SP)                               ;
                           B0  AD  9F  006C0      PUSHAB  INTMED_DATA                         ;
                           68  AE  9F  006C3      PUSHAB  CLASS_S_DESC                        ;
                00000000G  00      05  FB  006C6  CALLS   #5, OTS$CVT_T_D                     ;
```

```
                                 6E            50   D0 006CD          MOVL    R0, STATUS                                      ;
                                 65            6E   E8 006D0          BLBS    STATUS, 98$                                     ;
                                 50 00000000G  00   D0 006D3          MOVL    DBG$GL_OPCODE_NAME, R0                          ;
                                       30      AE   D5 006DA          TSTL    SCALE                                           ;
                                               0C   18 006DD          BGEQ    95$                                             ;
                                               50   DD 006DF          PUSHL   R0                                              ;
                                               01   DD 006E1          PUSHL   #1                                              ;
                                    0002869B   8F   DD 006E3          PUSHL   #165531                                         ;
                                               46   11 006E9          BRB     97$                                             ;
                                               50   DD 006EB 95$:     PUSHL   R0                                              ;
                                               01   DD 006ED          PUSHL   #1                                              ;
                                    00028A02   8F   DD 006EF          PUSHL   #166402                                         ;
                                               3A   11 006F5          BRB     97$                                             ;
                                 21            56   D1 006F7 96$:     CMPL    R6, #33                                         ; 2372
                                               3C   12 006FA          BNEQ    98$                                             ;
                              58 AE      FD      AD   B0 006FC          MOVW    SRC_INFO+5, CLASS_S_DESC                        ; 2374
                              5C AE      F9      AD   D0 00701          MOVL    SRC_INFO+1, CLASS_S_DESC+4                      ; 2375
                                 7E      55      8F   9A 00706          MOVZBL  #85, -(SP)                                      ; 2377
                                 7E      34      AE   CE 0070A          MNEGL   SCALE, -(SP)                                    ; 2376
                                 7E            D4   D4 0070E          CLRL    -(SP)                                           ;
                                         B0      AD   9F 00710          PUSHAB  INTMED_DATA                                     ;
                                         68      AE   9F 00713          PUSHAB  CLASS_S_DESC                                    ;
                           00000000G     00           FB 00716          CALLS   #5, OTS$CVT_T_D                                 ;
                                 6E            50   D0 0071D          MOVL    R0, STATUS                                      ;
                                 15            6E   E8 00720          BLBS    STATUS, 98$                                     ; 2378
                                    00000000G  00   DD 00723          PUSHL   DBG$GL_OPCODE_NAME                              ;
                                               01   DD 00729          PUSHL   #1                                              ;
                                    00028298   8F   DD 0072B          PUSHL   #164504                                         ;
                           00000000G     00           FB 00731 97$:     CALLS   #3, LIB$SIGNAL                                  ;
                        01                 0A           6B   8F 00738 98$:     CASEB   (R11), #10, #1                          ; 2382
                                       0064                005D      0073C 99$:     .WORD   107$-99$,-                              ;
                                                                                            108$-99$                            ;
                        01                 0C           6B   8F 00740          CASEB   (R11), #12, #1                          ; 2395
                                       04C8                0046      00744 100$:     .WORD   106$-100$,-                             ;
                                                                                            156$-100$                           ;
                                 03            56   D1 00748          CMPL    R6, #3                                          ; 2410
                                               08   12 0074B          BNEQ    101$                                            ;
                                    00000000'  EF   9F 0074D          PUSHAB  P.AEY                                           ; 2411
                                               32   11 00753          BRB     105$                                            ;
                                 09            56   D1 00755 101$:    CMPL    R6, #9                                          ; 2412
                                               08   12 00758          BNEQ    102$                                            ;
                                    00000000'  EF   9F 0075A          PUSHAB  P.AEZ                                           ; 2413
                                               25   11 00760          BRB     105$                                            ;
                                 0F            56   D1 00762 102$:    CMPL    R6, #15                                         ; 2414
                                               08   12 00765          BNEQ    103$                                            ;
                                    00000000'  EF   9F 00767          PUSHAB  P.AFA                                           ; 2415
                                               18   11 0076D          BRB     105$                                            ;
                                 1B            56   D1 0076F 103$:    CMPL    R6, #27                                         ; 2416
                                               08   12 00772          BNEQ    104$                                            ;
                                    00000000'  EF   9F 00774          PUSHAB  P.AFB                                           ; 2417
                                               0B   11 0077A          BRB     105$                                            ;
                                 21            56   D1 0077C 104$:    CMPL    R6, #33                                         ; 2418
                                               27   12 0077F          BNEQ    109$                                            ;
                                    00000000'  EF   9F 00781          PUSHAB  P.AFC                                           ; 2419
                                            0440   31 00787 105$:     BRW     151$                                            ;
                                 50       34    AE   D0 0078A 106$:     MOVL    OUTPUT, R0                                      ; 2400
                                 60       B0    AD   76 0078E          CVTDF   INTMED_DATA, (R0)                               ;
```

```
                04   A0        B8   AD   76 00792              CVTDF   INTMED_DATA+8, 4(R0)                    ;2401
                                   0F   11 00797              BRB     109$                                    ;2395
                34   BE        B0   AD   76 00799 107$:       CVTDF   INTMED_DATA, @OUTPUT                    ;2386
                                   08   11 0079E              BRB     109$
                     50        34   AE   D0 007A0 108$:       MOVL    OUTPUT, R0                              ;2390
                     60        B0   AD   7D 007A4              MOVQ    INTMED_DATA, (R0)
                                0467   31 007A8 109$:          BRW     157$                                    ;2424
                     04             56   D1 007AB 110$:       CMPL    R6, #4                                  ;2432
                                    71   12 007AE              BNEQ    115$
                     51        B0   AD   9E 007B0              MOVAB   INTMED_DATA, R1                         ;2433
                     50        B0   AD   9E 007B4              MOVAB   INTMED_DATA, R0
                          00000000G   00   16 007B8              JSB     DBG$CVT_CVTLH_R1
                                    58   D5 007BE 111$:       TSTL    BIN_SCALE
                                    15   15 007C0              BLEQ    112$
                     51        B0   AD   9E 007C2              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 007C6              MOVAB   P.AFD, R0
                          00000000G   00   16 007CD              JSB     DBG$CVT_MULH2_R1
                                    58   D7 007D3              DECL    BIN_SCALE
                                    E7   11 007D5              BRB     111$
                                    15   18 007D7 112$:       BGEQ    113$
                     51        B0   AD   9E 007D9              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 007DD              MOVAB   P.AFE, R0
                          00000000G   00   16 007E4              JSB     DBG$CVT_DIVH2_R1
                                    58   D6 007EA              INCL    BIN_SCALE
                                    E9   11 007EC              BRB     112$
                           30       AE   D5 007EE 113$:       TSTL    SCALE
                                    16   15 007F1              BLEQ    114$
                     51        B0   AD   9E 007F3              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 007F7              MOVAB   P.AFF, R0
                          00000000G   00   16 007FE              JSB     DBG$CVT_MULH2_R1
                                30   AE   D7 00804              DECL    SCALE
                                    E5   11 00807              BRB     113$
                                    7C   18 00809 114$:       BGEQ    119$
                     51        B0   AD   9E 0080B              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 0080F              MOVAB   P.AFG, R0
                          00000000G   00   16 00816              JSB     DBG$CVT_DIVH2_R1
                                30   AE   D6 0081C              INCL    SCALE
                                    E8   11 0081F              BRB     114$
                     0A             56   D1 00821 115$:       CMPL    R6, #10                                 ;2437
                                    7C   12 00824              BNEQ    121$
                     51      FF7C   CD   9E 00826              MOVAB   TEMP_BUF1, R1                           ;2438
                     50        B0   AD   9E 0082B              MOVAB   INTMED_DATA, R0
                          00000000G   00   16 0082F              JSB     DBG$CVT_CVTROUH_R1
B0   AD      FF7C   CD                  10   28 00835              MOVC3   #16, TEMP_BUF1, INTMED_DATA
                                    58   D5 0083C 116$:       TSTL    BIN_SCALE
                                    15   15 0083E              BLEQ    117$
                     51        B0   AD   9E 00840              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 00844              MOVAB   P.AFH, R0
                          00000000G   00   16 0084B              JSB     DBG$CVT_MULH2_R1
                                    58   D7 00851              DECL    BIN_SCALE
                                    E7   11 00853              BRB     116$
                                    15   18 00855 117$:       BGEQ    118$
                     51        B0   AD   9E 00857              MOVAB   INTMED_DATA, R1
                     50 00000000'   EF   9E 0085B              MOVAB   P.AFI, R0
                          00000000G   00   16 00862              JSB     DBG$CVT_DIVH2_R1
                                    58   D6 00868              INCL    BIN_SCALE
                                    E9   11 0086A              BRB     117$
```

```
                                30   AE   D5 0086C 118$:   TSTL    SCALE                                                  ;
                                16   15 0086F            BLEQ    119$                                                   ;
                             51   B0   AD 9E 00871        MOVAB   INTMED_DATA, R1                                        ;
                             50 00000000'  EF 9E 00875    MOVAB   P.AFJ, R0                                              ;
                                00000000G  00 16 0087C    JSB     DBG$CVT_MULH2_R1                                       ;
                                      30   AE   D7 00882  DECL    SCALE                                                  ;
                                           E5   11 00885  BRB     118$                                                   ;
                                           03   19 00887 119$:   BLSS    120$                                           ;
                                      02EF   31 00889      BRW     144$                                                   ;
                             51   B0   AD 9E 0088C 120$:   MOVAB   INTMED_DATA, R1                                     ;
                             50 00000000'  EF 9E 00890    MOVAB   P.AFK, R0                                              ;
                                00000000G  00 16 00897    JSB     DBG$CVT_DIVH2_R1                                       ;
                                      30   AE   D6 0089D  INCL    SCALE                                                  ;
                                           E5   11 008A0  BRB     119$                                                   ;
                             10            56   D1 008A2 121$:   CMPL    R6, #16                                        ; 2442
                                           03   13 008A5  BEQL    122$                                                   ;
                                      00CE   31 008A7      BRW     128$                                                   ;
                             51   FF7C   CD 9E 008AA 122$:   MOVAB   TEMP_BUF1, R1                                     ; 2443
                             50   B0   AD 9E 008AF        MOVAB   INTMED_DATA, R0                                        ;
                                00000000G  00 16 008B3    JSB     DBG$CVT_CVTDH_R1                                       ;
                             51   C0   AD 9E 008B9        MOVAB   INTMED_DATA+16, R1                                     ;
                             50   B8   AD 9E 008BD        MOVAB   INTMED_DATA+8, R0                                      ;
                                00000000G  00 16 008C1    JSB     DBG$CVT_CVTDH_R1                                       ;
B0   AD      FF7C   CD                   10   28 008C7  MOVC3   #16, TEMP_BUF1, INTMED_DATA                           ;
                                           58   D5 008CE 123$:   TSTL    BIN_SCALE                                      ;
                                           26   15 008D0  BLEQ    124$                                                   ;
                             51   B0   AD 9E 008D2        MOVAB   INTMED_DATA, R1                                        ;
                             50 00000000'  EF 9E 008D6    MOVAB   P.AFL, R0                                              ;
                                00000000G  00 16 008DD    JSB     DBG$CVT_MULH2_R1                                       ;
                             51   C0   AD 9E 008E3        MOVAB   INTMED_DATA+16, R1                                     ;
                             50 00000000'  EF 9E 008E7    MOVAB   P.AFM, R0                                              ;
                                00000000G  00 16 008EE    JSB     DBG$CVT_MULH2_R1                                       ;
                                           58   D7 008F4  DECL    BIN_SCALE                                              ;
                                           D6   11 008F6  BRB     123$                                                   ;
                                           26   18 008F8 124$:   BGEQ    125$                                           ;
                             51   B0   AD 9E 008FA        MOVAB   INTMED_DATA, R1                                        ;
                             50 00000000'  EF 9E 008FE    MOVAB   P.AFN, R0                                              ;
                                00000000G  00 16 00905    JSB     DBG$CVT_DIVH2_R1                                       ;
                             51   C0   AD 9E 0090B        MOVAB   INTMED_DATA+16, R1                                     ;
                             50 00000000'  EF 9E 0090F    MOVAB   P.AFO, R0                                              ;
                                00000000G  00 16 00916    JSB     DBG$CVT_DIVH2_R1                                       ;
                                           58   D6 0091C  INCL    BIN_SCALE                                              ;
                                           D8   11 0091E  BRB     124$                                                   ;
                                      30   AE   D5 00920 125$:   TSTL    SCALE                                          ;
                                           27   15 00923  BLEQ    126$                                                   ;
                             51   B0   AD 9E 00925        MOVAB   INTMED_DATA, R1                                        ;
                             50 00000000'  EF 9E 00929    MOVAB   P.AFP, R0                                              ;
                                00000000G  00 16 00930    JSB     DBG$CVT_MULH2_R1                                       ;
                             51   C0   AD 9E 00936        MOVAB   INTMED_DATA+16, R1                                     ;
                             50 00000000'  EF 9E 0093A    MOVAB   P.AFQ, R0                                              ;
                                00000000G  00 16 00941    JSB     DBG$CVT_MULH2_R1                                       ;
                                      30   AE   D7 00947  DECL    SCALE                                                  ;
                                           D4   11 0094A  BRB     125$                                                   ;
                                           03   19 0094C 126$:   BLSS    127$                                           ;
                                      022A   31 0094E      BRW     144$                                                   ;
                             51   B0   AD 9E 00951 127$:   MOVAB   INTMED_DATA, R1                                     ;
                             50 00000000'  EF 9E 00955    MOVAB   P.AFR, R0                                              ;
```

```
                00000000G   00  16  0095C           JSB     DBG$CVT_DIVH2_R1                                        ;
                      51    CO  AD  9E  00962       MOVAB   INTMED_DATA+16, R1                                      ;
                      50 00000000'  EF  9E  00966   MOVAB   P.AFS, R0                                              ;
                00000000G   00  16  0096D           JSB     DBG$CVT_DIVH2_R1                                        ;
                      30    AE  D6  00973           INCL    SCALE                                                   ;
                            D4  11  00976           BRB     126$                                                    ;
                      16        56  D1  00978 128$: CMPL    R6, #22                                                 ; 2447
                                03  13  0097B       BEQL    129$                                                    ;
                              0186  31  0097D       BRW     140$                                                    ;
                      1B    04  BE  91  00980 129$: CMPB    @4(SP), #27                                             ; 2449
                                09  13  00984       BEQL    130$                                                    ;
                      1D    04  BE  91  00986       CMPB    @4(SP), #29                                             ; 2450
                                03  13  0098A       BEQL    130$                                                    ;
                             00CB  31  0098C       BRW     135$                                                    ;
                      51  FF7C  CD  9E  0098F 130$: MOVAB   TEMP_BUF1, R1                                          ; 2451
                      50    B0  AD  9E  00994       MOVAB   INTMED_DATA, R0                                         ;
                00000000G   00  16  00998           JSB     DBG$CVT_CVTGH_R1                                        ;
                      51    C0  AD  9E  0099E       MOVAB   INTMED_DATA+16, R1                                      ;
                      50    B8  AD  9E  009A2       MOVAB   INTMED_DATA+8, R0                                       ;
                00000000G   00  16  009A6           JSB     DBG$CVT_CVTGH_R1                                        ;
B0  AD  FF7C    CD              10  28  009AC       MOVC3   #16, TEMP_BUF1, INTMED_DATA                             ;
                            58  D5  009B3 131$:     TSTL    BIN_SCALE                                               ;
                            26  15  009B5           BLEQ    132$                                                    ;
                      51    B0  AD  9E  009B7       MOVAB   INTMED_DATA, R1                                         ;
                      50 00000000'  EF  9E  009BB   MOVAB   P.AFT, R0                                               ;
                00000000G   00  16  009C2           JSB     DBG$CVT_MULH2_R1                                        ;
                      51    C0  AD  9E  009C8       MOVAB   INTMED_DATA+16, R1                                      ;
                      50 00000000'  EF  9E  009CC   MOVAB   P.AFU, R0                                               ;
                00000000G   00  16  009D3           JSB     DBG$CVT_MULH2_R1                                        ;
                            58  D7  009D9           DECL    BIN_SCALE                                               ;
                            D6  11  009DB           BRB     131$                                                    ;
                            26  18  009DD 132$:     BGEQ    133$                                                    ;
                      51    B0  AD  9E  009DF       MOVAB   INTMED_DATA, R1                                         ;
                      50 00000000'  EF  9E  009E3   MOVAB   P.AFV, R0                                               ;
                00000000G   00  16  009EA           JSB     DBG$CVT_DIVH2_R1                                        ;
                      51    C0  AD  9E  009F0       MOVAB   INTMED_DATA+16, R1                                      ;
                      50 00000000'  EF  9E  009F4   MOVAB   P.AFW, R0                                               ;
                00000000G   00  16  009FB           JSB     DBG$CVT_DIVH2_R1                                        ;
                            58  D6  00A01           INCL    BIN_SCALE                                               ;
                            D8  11  00A03           BRB     132$                                                    ;
                      30    AE  D5  00A05 133$:     TSTL    SCALE                                                   ;
                            27  15  00A08           BLEQ    134$                                                    ;
                      51    B0  AD  9E  00A0A       MOVAB   INTMED_DATA, R1                                         ;
                      50 00000000'  EF  9E  00A0E   MOVAB   P.AFX, R0                                               ;
                00000000G   00  16  00A15           JSB     DBG$CVT_MULH2_R1                                        ;
                      51    C0  AD  9E  00A1B       MOVAB   INTMED_DATA+16, R1                                      ;
                      50 00000000'  EF  9E  00A1F   MOVAB   P.AFY, R0                                               ;
                00000000G   00  16  00A26           JSB     DBG$CVT_MULH2_R1                                        ;
                      30    AE  D7  00A2C           DECL    SCALE                                                   ;
                            D4  11  00A2F           BRB     133$                                                    ;
                            51  18  00A31 134$:     BGEQ    136$                                                    ;
                      51    B0  AD  9E  00A33       MOVAB   INTMED_DATA, R1                                         ;
                      50 00000000'  EF  9E  00A37   MOVAB   P.AFZ, R0                                               ;
                00000000G   00  16  00A3E           JSB     DBG$CVT_DIVH2_R1                                        ;
                      51    C0  AD  9E  00A44       MOVAB   INTMED_DATA+16, R1                                      ;
                      50 00000000'  EF  9E  00A48   MOVAB   P.AGA, R0                                               ;
                00000000G   00  16  00A4F           JSB     DBG$CVT_DIVH2_R1                                        ;
```

```
                                30   AE  D6 00A55          INCL    SCALE                                                   ;
                                     D7  11 00A58          BRB     134$                                                    ;
                                     58  D5 00A5A 135$:    TSTL    BIN_SCALE                                               ; 2453
                                     26  15 00A5C          BLEQ    136$                                                    ;
                                51   B0  AD 9E 00A5E       MOVAB   INTMED_DATA, R1                                         ;
                                50 00000000'  EF 9E 00A62  MOVAB   P.AGB, R0                                               ;
                                   00000000G  00 16 00A69  JSB     DBG$CVT_MULH2_R1                                        ;
                                51   C0  AD 9E 00A6F       MOVAB   INTMED_DATA+16, R1                                      ;
                                50 00000000'  EF 9E 00A73  MOVAB   P.AGC, R0                                               ;
                                   00000000G  00 16 00A7A  JSB     DBG$CVT_MULH2_R1                                        ;
                                     58  D7 00A80          DECL    BIN_SCALE                                               ;
                                     D6  11 00A82          BRB     135$                                                    ;
                                     58  D5 00A84 136$:    TSTL    BIN_SCALE                                               ;
                                     26  18 00A86          BGEQ    137$                                                    ;
                                51   B0  AD 9E 00A88       MOVAB   INTMED_DATA, R1                                         ;
                                50 00000000'  EF 9E 00A8C  MOVAB   P.AGD, R0                                               ;
                                   00000000G  00 16 00A93  JSB     DBG$CVT_DIVH2_R1                                        ;
                                51   C0  AD 9E 00A99       MOVAB   INTMED_DATA+16, R1                                      ;
                                50 00000000'  EF 9E 00A9D  MOVAB   P.AGE, R0                                               ;
                                   00000000G  00 16 00AA4  JSB     DBG$CVT_DIVH2_R1                                        ;
                                     58  D6 00AAA          INCL    BIN_SCALE                                               ;
                                     D6  11 00AAC          BRB     136$                                                    ;
                                30   AE  D5 00AAE 137$:    TSTL    SCALE                                                   ;
                                     27  15 00AB1          BLEQ    138$                                                    ;
                                51   B0  AD 9E 00AB3       MOVAB   INTMED_DATA, R1                                         ;
                                50 00000000'  EF 9E 00AB7  MOVAB   P.AGF, R0                                               ;
                                   00000000G  00 16 00ABE  JSB     DBG$CVT_MULH2_R1                                        ;
                                51   C0  AD 9E 00AC4       MOVAB   INTMED_DATA+16, R1                                      ;
                                50 00000000'  EF 9E 00AC8  MOVAB   P.AGG, R0                                               ;
                                   00000000G  00 16 00ACF  JSB     DBG$CVT_MULH2_R1                                        ;
                                     30  AE  D7 00AD5          DECL    SCALE                                               ;
                                     D4  11 00AD8          BRB     137$                                                    ;
                                     03  19 00ADA 138$:    BLSS    139$                                                    ;
                                   009C  31 00ADC          BRW     144$                                                    ;
                                51   B0  AD 9E 00ADF 139$:  MOVAB   INTMED_DATA, R1                                     ;
                                50 00000000'  EF 9E 00AE3  MOVAB   P.AGH, R0                                               ;
                                   00000000G  00 16 00AEA  JSB     DBG$CVT_DIVH2_R1                                        ;
                                51   C0  AD 9E 00AF0       MOVAB   INTMED_DATA+16, R1                                      ;
                                50 00000000'  EF 9E 00AF4  MOVAB   P.AGI, R0                                               ;
                                   00000000G  00 16 00AFB  JSB     DBG$CVT_DIVH2_R1                                        ;
                                     30  AE  D6 00B01          INCL    SCALE                                               ;
                                     D4  11 00B04          BRB     138$                                                    ;
                                1C             56  D1 00B06 140$:  CMPL    R6, #28                                         ; 2457
                                               70  12 00B09       BNEQ    144$                                            ;
                           2C   AE      FD     AD  3C 00B0B       MOVZWL  SRC_INFO+5, NO_DIGITS                           ; 2458
                           09   03      AA     91  00B10          CMPB    3(R10), #9                                      ;
                                        0A     12  00B14          BNEQ    141$                                            ;
                           30   AE      08     AA  98 00B16       CVTBL   8(R10), SCALE                                   ;
                           30   AE      30     AE  CE 00B1B       MNEGL   SCALE, SCALE                                    ;
FF7C  CD   2C   AE   B0    AD      2C     AE  08  00B20 141$:    CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -             ;
                                                                         TEMP_BUF1                                       ;
           58   AE   2C    AE                 01  A1 00B2A       ADDW3   #1, NO_DIGITS, CLASS_S_DESC                      ;
                     5C    AE      FF7C   CD  9E 00B30          MOVAB   TEMP_BUF1, CLASS_S_DESC+4                       ;
                     7E    44      8F     9A  00B36          MOVZBL  #68, -(SP)                                          ;
                           34      AE     DD  00B3A          PUSHL   SCALE                                               ;
                                          7E  D4 00B3D       CLRL    -(SP)                                               ;
                                   B0     AD  9F 00B3F       PUSHAB  INTMED_DATA                                         ;
```

```
                                68  AE  9F 00B42          PUSHAB  CLASS_S_DESC                                    ;
                00000000G       00  05  FB 00B45          CALLS   #5, OTS$CVT_T_H                                 ;
                                6E  50  D0 00B4C          MOVL    R0, STATUS                                      ;
                                29  6E  E8 00B4F          BLBS    STATUS, 144$                                    ;
                                50 00000000G 00 D0 00B52  MOVL    DBG$GL_OPCODE_NAME, R0                          ;
                                    30  AE  D5 00B59      TSTL    SCALE                                           ;
                                    0C  18 00B5C          BGEQ    142$                                            ;
                                    50  DD 00B5E          PUSHL   R0                                              ;
                                    01  DD 00B60          PUSHL   #1                                              ;
                                0002869B    8F  DD 00B62  PUSHL   #165531                                         ;
                                    0A  11 00B68          BRB     143$                                            ;
                                    50  DD 00B6A 142$:     PUSHL   R0                                              ;
                                    01  DD 00B6C          PUSHL   #1                                              ;
                                00028A02    8F  DD 00B6E  PUSHL   #166402                                         ;
                00000000G       00      03  FB 00B74 143$:  CALLS   #3, LIB$SIGNAL                                ;
01                          1B      6B  8F 00B7B 144$:  CASEB   (R11), #27, #1                                    ; 2463
                        008D        007D    00B7F 145$:  .WORD   154$-145$,-                                       ;
                                                         156$-145$                                              ;
01                          1D      6B  8F 00B83          CASEB   (R11), #29, #1                                  ; 2473
                        006D        0054    00B87 146$:  .WORD   152$-146$,-                                       ;
                                                         153$-146$                                              ;
                                04      56  D1 00B8B          CMPL    R6, #4                                      ; 2488
                                    08  12 00B8E          BNEQ    147$                                            ;
                                00000000'   EF  9F 00B90  PUSHAB  P.AGJ                                           ; 2489
                                    32  11 00B96          BRB     151$                                            ;
                                0A      56  D1 00B98 147$:  CMPL    R6, #10                                     ; 2490
                                    08  12 00B9B          BNEQ    148$                                            ;
                                00000000'   EF  9F 00B9D  PUSHAB  P.AGK                                           ; 2491
                                    25  11 00BA3          BRB     151$                                            ;
                                10      56  D1 00BA5 148$:  CMPL    R6, #16                                     ; 2492
                                    08  12 00BA8          BNEQ    149$                                            ;
                                00000000'   EF  9F 00BAA  PUSHAB  P.AGL                                           ; 2493
                                    18  11 00BB0          BRB     151$                                            ;
                                16      56  D1 00BB2 149$:  CMPL    R6, #22                                     ; 2494
                                    08  12 00BB5          BNEQ    150$                                            ;
                                00000000'   EF  9F 00BB7  PUSHAB  P.AGM                                           ; 2495
                                    0B  11 00BBD          BRB     151$                                            ;
                                1C      56  D1 00BBF 150$:  CMPL    R6, #28                                     ; 2496
                                    4E  12 00BC2          BNEQ    157$                                            ;
                                00000000'   EF  9F 00BC4  PUSHAB  P.AGN                                           ; 2497
                                    01  DD 00BCA 151$:  PUSHL   #1                                              ;
                                00028362    8F  DD 00BCC  PUSHL   #164706                                         ;
                00000000G       00      03  FB 00BD2          CALLS   #3, LIB$SIGNAL                              ;
                                    37  11 00BD9          BRB     157$                                            ; 2486
                                50      B0  AD 9E 00BDB 152$:  MOVAB   INTMED_DATA, R0                            ; 2478
                                51      34  AE D0 00BDF          MOVL    OUTPUT, R1                                ;
                                00000000G   00  16 00BE3  JSB     DBG$CVT_CVTHG_R1                                ;
        51          34      AE              08  C1 00BE9          ADDL3   #8, OUTPUT, R1                          ; 2479
                                50      C0  AD 9E 00BEE          MOVAB   INTMED_DATA+16, R0                      ;
                                    10  11 00BF2          BRB     155$                                            ;
34      BE          B0      AD              20  28 00BF4 153$:  MOVC3   #32, INTMED_DATA, @OUTPUT                ; 2483
                                    16  11 00BFA          BRB     157$                                            ; 2473
                                50      B0  AD 9E 00BFC 154$:  MOVAB   INTMED_DATA, R0                            ; 2467
                                51      34  AE D0 00C00          MOVL    OUTPUT, R1                                ;
                                00000000G   00  16 00C04 155$:  JSB     DBG$CVT_CVTHG_R1                        ;
                                    06  11 00C0A          BRB     157$                                            ;
34      BE          B0      AD              10  28 00C0C 156$:  MOVC3   #16, INTMED_DATA, @OUTPUT                ; 2470
```

```
                                06      FF    AD   E9 00C12 157$:    BLBC    SRC_INFO+7, 158$                                ; 2502
                            34  BE    8000  8F   A8 00C16           BISW2   #32768, @OUTPUT
                                      23B9  31   00C1C 158$:         BRW     649$                                            ; 2187
                                05          56   D1 00C1F 159$:     CMPL    R6, #5                                          ; 2510
                                            03   13 00C22           BEQL    160$
                                            00C2 31 00C24           BRW     168$
                                      B0    AD   D5 00C27 160$:     TSTL    INTMED_DATA                                     ; 2511
                                            04   18 00C2A           BGEQ    161$
                                FF  AD            01   88 00C2C     BISB2   #1, SRC_INFO+7
                                2C  AE            1F   D0 00C30 161$:     MOVL    #31, NO_DIGITS
                          B0    AD    2C  AE      B0   AD F9 00C34        CVTLP   INTMED_DATA, NO_DIGITS, INTMED_DATA
                                      30  AE      D5   00C3B              TSTL    SCALE
                                      3C  13      00C3E                   BEQL    164$
                      FF7C    CD    B0  AD        2C   AE 34 00C40       MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1
                                0E    0C  AE      E9   00C48              BLBC    CVT_ROUND_FLAG, 162$
                                55    B0  AD      9E   00C4C              MOVAB   INTMED_DATA, R5
                                54    2C  AE      9E   00C50              MOVAB   NO_DIGITS, R4
                                08  AE            05   D0 00C54          MOVL    #5, 8(SP)
                                            0B   11 00C58           BRB     163$
                                55    B0  AD      9E   00C5A 162$:     MOVAB   INTMED_DATA, R5
                                54    2C  AE      9E   00C5E           MOVAB   NO_DIGITS, R4
                                      08  AE      D4   00C62           CLRL    8(SP)
                                53    08  AE      9E   00C65 163$:     MOVAB   8(SP), R3
                                52  FF7C  CD      9E   00C69           MOVAB   TEMP_BUF1, R2
                                51    2C  AE      9E   00C6E           MOVAB   NO_DIGITS, R1
                                50    30  AE      9E   00C72           MOVAB   SCALE, R0
                                  00000000G   00   16 00C76           JSB     DBG$CVT_ASHP_R1
                                            58   D5 00C7C 164$:     TSTL    BIN_SCALE
                                            32   15 00C7E           BLEQ    165$
                      FF7C    CD    B0  AD        2C   AE 34 00C80       MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1
                                55    B0  AD      9E   00C88              MOVAB   INTMED_DATA, R5
                                54    2C  AE      9E   00C8C              MOVAB   NO_DIGITS, R4
                                53  FF7C  CD      9E   00C90              MOVAB   TEMP_BUF1, R3
                                52    2C  AE      9E   00C95              MOVAB   NO_DIGITS, R2
                                51  00000000'   EF   9E 00C99           MOVAB   P.AGO, R1
                                08  AE            01   D0 00CA0          MOVL    #1, 8(SP)
                                50    08  AE      9E   00CA4              MOVAB   8(SP), R0
                                  00000000G   00   16 00CA8           JSB     DBG$CVT_MULP_R1
                                            58   D7 00CAE           DECL    BIN_SCALE
                                            CA   11 00CB0           BRB     164$
                                            03   19 00CB2 165$:     BLSS    167$
                                          0105   31 00CB4 166$:     BRW     175$
                      FF7C    CD    B0  AD        2C   AE 34 00CB7 167$:    MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1
                                55    B0  AD      9E   00CBF              MOVAB   INTMED_DATA, R5
                                54    2C  AE      9E   00CC3              MOVAB   NO_DIGITS, R4
                                53  FF7C  CD      9E   00CC7              MOVAB   TEMP_BUF1, R3
                                52    2C  AE      9E   00CCC              MOVAB   NO_DIGITS, R2
                                51  00000000'   EF   9E 00CD0           MOVAB   P.AGP, R1
                                08  AE            01   D0 00CD7          MOVL    #1, 8(SP)
                                50    08  AE      9E   00CDB              MOVAB   8(SP), R0
                                  00000000G   00   16 00CDF           JSB     DBG$CVT_DIVP_R1
                                            58   D6 00CE5           INCL    BIN_SCALE
                                            C9   11 00CE7           BRB     165$
                                1D                56   D1 00CE9 168$:     CMPL    R6, #29                                 ; 2515
                                            C6   12 00CEC           BNEQ    166$
                                2C  AE        FD  AD 3C 00CEE           MOVZWL  SRC_INFO+5, NO_DIGITS                   ; 2516
  08  BE            01    B0  AD        2C   AE 37 00CF3              CMPP4   NO_DIGITS, INTMED_DATA, #1, @PACK_ZERO
```

```
                                                 54  DC  00CFB          MOVPSL  R4
        54              54              02       02  EF  00CFD          EXTZV   #2, #2, R4, R4
                                                 54  D7  00D02          DECL    R4
                                                 04  15  00D04          BLEQ    169$
                                FF      AD       01  88  00D06          BISB2   #1, SRC_INFO+7
                                        30   AE  D5  00D0A 169$:        TSTL    SCALE
                                             3C  13  00D0D              BEQL    172$
                    FF7C    CD      B0   AD   2C   AE  34  00D0F        MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1
                                         0E  0C   AE  E9  00D17        BLBC    CVT_ROUND_FLAG, 170$
                                             55   B0  AD  9E  00D1B    MOVAB   INTMED_DATA, R5
                                             54   2C  AE  9E  00D1F    MOVAB   NO_DIGITS, R4
                                        08   AE       05  D0  00D23    MOVL    #5, 8(SP)
                                                 0B  11  00D27          BRB     171$
                                             55   B0  AD  9E  00D29 170$:  MOVAB   INTMED_DATA, R5
                                             54   2C  AE  9E  00D2D    MOVAB   NO_DIGITS, R4
                                                  08  AE  D4  00D31    CLRL    8(SP)
                                             53   08  AE  9E  00D34 171$:  MOVAB   8(SP), R3
                                             52  FF7C CD  9E  00D38    MOVAB   TEMP_BUF1, R2
                                             51   2C  AE  9E  00D3D    MOVAB   NO_DIGITS, R1
                                             50   30  AE  9E  00D41    MOVAB   SCALE, R0
                                        00000000G    00  16  00D45    JSB     DBG$CVT_ASHP_R1
                                             50       6A  3C  00D4B 172$:  MOVZWL  (R10), R0
                                             51   08  AA  98  00D4E    CVTBL   8(R10), R1
                                             50       51  C0  00D52    ADDL2   R1, R0
                                             52       69  3C  00D55    MOVZWL  (R9), R2
                                             51   08  A9  98  00D58    CVTBL   8(R9), R1
                                             52       51  C0  00D5C    ADDL2   R1, R2
                                             52       50  D1  00D5F    CMPL    R0, R2
                                                      58  15  00D62    BLEQ    175$
                                             15   04  BE  91  00D64    CMPB    @4(SP), #21
                                                      52  12  00D68    BNEQ    175$
                                             15       6B  91  00D6A    CMPB    (R11), #21
                                                      4D  12  00D6D    BNEQ    175$
                                             50   BF  AD  9E  00D6F    MOVAB   INTMED_DATA+15, HIGH_NIBBLE_PTR
                                             52       69  3C  00D73    MOVZWL  (R9), R2
                                             52       02  C6  00D76    DIVL2   #2, R2
                                             52       50  C2  00D79    SUBL2   HIGH_NIBBLE_PTR, R2
                                             52       52  CE  00D7C    MNEGL   R2, LOW_NIBBLE_PTR
                                             50       69  3C  00D7F    MOVZWL  (R9), R0
        7E              00                   50       01  7A  00D82    EMUL    #1, R0, #0, -(SP)
        50              50                   8E       02  7B  00D87    EDIV    #2, (SP)+, R0, R0
                                                      50  D5  00D8C    TSTL    R0
                                                      08  12  00D8E    BNEQ    173$
        50              62                   04       00  EF  00D90    EXTZV   #0, #4, (LOW_NIBBLE_PTR), R0
                                             62       50  90  00D95    MOVB    R0, (LOW_NIBBLE_PTR)
                                                      52  D7  00D98 173$:  DECL    LOW_NIBBLE_PTR
                                             50   B0  AD  9E  00D9A    MOVAB   INTMED_DATA, R0
                                             50       52  D1  00D9E    CMPL    LOW_NIBBLE_PTR, R0
                                                      04  19  00DA1    BLSS    174$
                                                      62  94  00DA3    CLRB    (LOW_NIBBLE_PTR)
                                                      F1  11  00DA5    BRB     173$
                                        00000000G     00  DD  00DA7 174$:  PUSHL   DBG$GL_OPCODE_NAME
                                                      01  DD  00DAD    PUSHL   #1
                                        0002809B      8F  DD  00DAF    PUSHL   #163995
                                00000000G   00        03  FB  00DB5    CALLS   #3, LIB$SIGNAL
                            06              0F        6B  8F  00DBC 175$:  CASEB   (R11), #15, #6                                  2521
    009F            0069            0051        0029      00DC0 176$:  .WORD   179$-176$,-
```

```
                    00ED          00CD          00CD     00DC8                    181$-176$,-                                                    ;
                                                                                 184$-176$,-                                                    ;
                                                                                 188$-176$,-                                                    ;
                                                                                 192$-176$,-                                                    ;
                                                                                 192$-176$,-                                                    ;
                                                                                 196$-176$                                                      ;
                                 05                 56  D1 00DCE                CMPL    R6, #5                                                 ; 2574
                                                    08  12 00DD1                BNEQ    177$                                                   ;
                                    00000000'       EF  9F 00DD3                PUSHAB  P.AGQ                                                  ; 2575
                                                    0B  11 00DD9                BRB     178$                                                   ;
                                 1D                 56  D1 00DDB 177$:         CMPL    R6, #29                                                ; 2576
                                                    7D  12 00DDE                BNEQ    187$                                                   ;
                                    00000000'       EF  9F 00DE0                PUSHAB  P.AGR                                                  ; 2577
                                              21E0  31  00DE6 178$:             BRW     647$                                                   ;
                                 15       FF        AD  E9 00DE9 179$:          BLBC    SRC_INFO+7, 180$                                       ; 2526
                                    00000000G       00  DD 00DED                PUSHL   DBG$GL_OPCODE_NAME                                     ;
                                                    01  DD 00DF3                PUSHL   #1                                                     ;
                                    00028EF0        8F  DD 00DF5                PUSHL   #167664                                                ;
                           00000000G 00             03  FB 00DFB                CALLS   #3, LIB$SIGNAL                                         ;
          69 00000000G  00       B0  AD       2C    AE  24 00E02 180$:          CVTPT   NO_DIGITS, INTMED_DATA, LIB$AB_CVTPT_U, -              ; 2527
                                    34          BE  00E0D                       (R9), @OUTPUT                                                  ;
                                                7A  11 00E0F                BRB     191$                                                   ; 2521
                                                69  B5 00E11 181$:          TSTW    (R9)                                                   ; 2533
                                                04  12 00E13                BNEQ    182$                                                   ;
                                                50  D4 00E15                CLRL    R0                                                     ;
                                                05  11 00E17                BRB     183$                                                   ;
                                 50                 69  3C 00E19 182$:          MOVZWL  (R9), R0                                               ;
                                                    50  D7 00E1C                DECL    R0                                                     ;
    34    BE            50       B0  AD       2C    AE  08 00E1E 183$:          CVTPS   NO_DIGITS, INTMED_DATA, R0, @OUTPUT                    ; 2534
                                              0082  31  00E26                BRW     195$                                                   ; 2531
          69 00000000G  00       B0  AD       2C    AE  24 00E29 184$:          CVTPT   NO_DIGITS, INTMED_DATA, LIB$AB_CVTPT_U, -              ; 2538
                                                FF7C    CD  00E34                   (R9), TEMP_BUF1                                            ;
                                 50             FF7C    CD  9A 00E37                MOVZBL  TEMP_BUF1, R0                                      ; 2539
                                 50 00000000G00040      9E  00E3C                MOVAB   LIB$AB_CVT_U_0-48[R0], R0                          ;
                                 06                FF   AD  E9 00E44                BLBC    SRC_INFO+7, T85$                                   ;
                                 50                0A   A0  D0 00E48                MOVL    10(R0), R0                                         ;
                                                        03  11 00E4C                BRB     186$                                               ;
                                 50                     60  D0 00E4E 185$:          MOVL    (R0), R0                                       ; 2540
                                              FF7C      CD  50  90 00E51 186$:      MOVB    R0, TEMP_BUF1                                  ; 2539
                        34    BE              FF7C      CD  69  28 00E56            MOVC3   (R9), TEMP_BUF1, @OUTPUT                       ; 2541
                                                          61  11 00E5D 187$:        BRB     197$                                           ; 2521
                                                          69  B5 00E5F 188$:        TSTW    (R9)                                           ; 2550
                                                          04  12 00E61              BNEQ    189$                                           ;
                                                          57  D4 00E63              CLRL    DES_LEN                                        ;
                                                          05  11 00E65              BRB     190$                                           ;
                                 57                       69  3C 00E67 189$:        MOVZWL  (R9), DES_LEN                                  ;
                                                          57  D7 00E6A              DECL    DES_LEN                                        ;
FF7C    CD                57       B0  AD       2C    AE  08 00E6C 190$:          CVTPS   NO_DIGITS, INTMED_DATA, DES_LEN, TEMP_BUF1             ; 2552
                                    B0  AD47        FF7C    CD  90 00E75            MOVB    TEMP_BUF1, INTMED_DATA[DES_LEN]                ; 2553
                        B0    AD        FF7D      CD  57  28 00E7C            MOVC3   DES_LEN, TEMP_BUF1+1, INTMED_DATA                      ; 2554
                                                          57  D6 00E83              INCL    R7                                             ; 2555
                        34    BE        B0        AD  57  28 00E85            MOVC3   R7, INTMED_DATA, @OUTPUT                               ;
                                                          33  11 00E88 191$:        BRB     197$                                           ; 2521
                                 13                       68  91 00E8D 192$:        CMPB    (R11), #19                                     ; 2560
                                                          09  12 00E90              BNEQ    193$                                           ;
                                 50 00000000G             00  9E 00E92              MOVAB   LIB$AB_CVTPT_0, R0                             ;
                                                          07  11 00E99              BRB     194$                                           ;
```

```
                                                50 00000000G   00    9E  00E9B 193$:    MOVAB   LIB$AB_CVTPT_Z, R0
            69              60          B0      AD         2C   AE    24  00EA2 194$:    CVTPT   NO_DIGITS, INTMED_DATA, (R0), (R9), @OUTPUT    2561
                                                           34   BE          00EA9
                                                           13   11    00EAB 195$:    BRB     197$
     FF7C   CD          2C   AE          B0      AD         2C   AE    08  00EAD 196$:    CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -           2567
                                                                                               TEMP_BUF1
       34   BE              69      FF7C    CD         2C   AE    09  00EB7          CVTSP   NO_DIGITS, TEMP_BUF1, (R9), @OUTPUT             2568
                                                      2115   31    00EC0 197$:    BRW     649$                                                2187
                                                           30   AE    D5  00EC3 198$:    TSTL    SCALE                                           2583
                                                           0F   15    00EC6          BLEQ    199$
                                                50         B0   AD    9E  00EC8          MOVAB   INTMED_DATA, R0
                                                00000000G  00   16    00ECC          JSB     LIB$$CVT_SCALE_OU_UP_BY_10_R1
                                                           30   AE    D7  00ED2          DECL    SCALE
                                                           EC   11    00ED5          BRB     198$
                                                           0F   18    00ED7 199$:    BGEQ    200$
                                                50         B0   AD    9E  00ED9          MOVAB   INTMED_DATA, R0
                                                00000000G  00   16    00EDD          JSB     LIB$$CVT_SCALE_OU_DOWN_BY_10_R1
                                                           30   AE    D6  00EE3          INCL    SCALE
                                                           EF   11    00EE6          BRB     199$
                                                           58   D5    00EE8 200$:    TSTL    BIN_SCALE
                                                           0E   15    00EEA          BLEQ    201$
                                                50         B0   AD    9E  00EEC          MOVAB   INTMED_DATA, R0
                                                00000000G  00   16    00EF0          JSB     DBG$CVT_SCALE_OU_UP_BY_2_R1
                                                           58   D7    00EF6          DECL    BIN_SCALE
                                                           EE   11    00EF8          BRB     200$
                                                           0E   18    00EFA 201$:    BGEQ    202$
                                                50         B0   AD    9E  00EFC          MOVAB   INTMED_DATA, R0
                                                00000000G  00   16    00F00          JSB     DBG$CVT_SCALE_OU_DOWN_BY_2_R1
                                                           58   D6    00F06          INCL    BIN_SCALE
                                                           F0   11    00F08          BRB     201$
                            50          B4      AD         B8   AD    C9  00F0A 202$:    BISL3   INTMED_DATA+8, INTMED_DATA+4, R0            2585
                                                50         BC   AD    C8  00F10          BISL2   INTMED_DATA+12, R0
                                                           15   13    00F14          BEQL    203$
                                                00000000G  00   DD    00F16          PUSHL   DBG$GL_OPCODE_NAME                                  2587
                                                           01   DD    00F1C          PUSHL   #1
                                                000286A3   8F   DD    00F1E          PUSHL   #165539
                                      00000000G  00         03   FB    00F24          CALLS   #3, LIB$SIGNAL
                                              29           01   6B    8F  00F2B 203$:    CASEB   (R11), #1, #41                                  2588
                                0054          0082         005D        0130  00F2F 204$:    .WORD   223$-204$,-
                                00F8          00CE         00A4        0054  00F37          206$-204$,-
                                0054          0054         0054        0054  00F3F          209$-204$,-
                                0054          0054         0054        0054  00F47          205$-204$,-
                                0054          0054         0054        0054  00F4F          205$-204$,-
                                0054          0054         0054        0054  00F57          212$-204$,-
                                0054          0054         0054        0054  00F5F          215$-204$,-
                                0054          0054         0054        0054  00F67          218$-204$,-
                                0054          0054         0130        0054  00F6F          205$-204$,-
                                0130          0054         0054        0054  00F77          205$-204$,-
                                                           0130        0130  00F7F          205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
                                                                                           205$-204$,-
```

```
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 223$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 205$-204$,-                              :
                                                                 223$-204$,-                              :
                                                                 223$-204$,-                              :
                                                                 223$-204$                                :
                        00000000'  EF   9F  00F83  205$:  PUSHAB  P.AGS                                   : 2644
                             203D  31  00F89              BRW     647$                                    :
              05          B1  AD   E8  00F8C  206$:  BLBS    INTMED_DATA+1, 207$                          : 2593
                          B2  AD   B5  00F90         TSTW    INTMED_DATA+2                                :
                              15   13  00F93         BEQL    208$                                         :
                    00000000G  00  DD  00F95  207$:  PUSHL   DBG$GL_OPCODE_NAME                           : 2594
                               01  DD  00F9B         PUSHL   #1                                           :
                    000286A3   8F  DD  00F9D         PUSHL   #165539                                      :
  00000000G   00               03  FB  00FA3         CALLS   #3, LIB$SIGNAL                               :
         34   BE          B0  AD   90  00FAA  208$:  MOVB    INTMED_DATA, @OUTPUT                         : 2595
                              1F   11  00FAF         BRB     211$                                         : 2588
                          B2  AD   B5  00FB1  209$:  TSTW    INTMED_DATA+2                                : 2600
                              15   13  00FB4         BEQL    210$                                         :
                    00000000G  00  DD  00FB6         PUSHL   DBG$GL_OPCODE_NAME                           :
                               01  DD  00FBC         PUSHL   #1                                           :
                    000286A3   8F  DD  00FBE         PUSHL   #165539                                      :
  00000000G   00               03  FB  00FC4         CALLS   #3, LIB$SIGNAL                               :
         34   BE          B0  AD   B0  00FCB  210$:  MOVW    INTMED_DATA, @OUTPUT                         : 2601
                            008A   31  00FD0  211$:  BRW     222$                                         : 2588
                          B0  AD   D5  00FD3  212$:  TSTL    INTMED_DATA                                  : 2606
                              15   18  00FD6         BGEQ    213$                                         :
                    00000000G  00  DD  00FD8         PUSHL   DBG$GL_OPCODE_NAME                           :
                               01  DD  00FDE         PUSHL   #1                                           :
                    000286A3   8F  DD  00FE0         PUSHL   #165539                                      :
  00000000G   00               03  FB  00FE6         CALLS   #3, LIB$SIGNAL                               :
              05          FF  AD   E9  00FED  213$:  BLBC    SRC_INFO+7, 214$                             : 2607
         B0   AD          B0  AD   CE  00FF1         MNEGL   INTMED_DATA, INTMED_DATA                     :
              50          B0  AD   9E  00FF6  214$:  MOVAB   INTMED_DATA, R0                              : 2608
                            18FB   31  00FFA         BRW     540$                                         :
                          B0  AD   D5  00FFD  215$:  TSTL    INTMED_DATA                                  : 2613
                              15   18  01000         BGEQ    216$                                         :
                    00000000G  00  DD  01002         PUSHL   DBG$GL_OPCODE_NAME                           :
                               01  DD  01008         PUSHL   #1                                           :
                    000286A3   8F  DD  0100A         PUSHL   #165539                                      :
```

```
                                00000000G  00          03  FB 01010          CALLS   #3, LIB$SIGNAL                          ;
                                        05      FF      AD  E9 01017 216$:    BLBC    SRC_INFO+7, 217$                        ; 2614
                                     B0 AD      B0      AD  CE 0101B          MNEGL   INTMED_DATA, INTMED_DATA                ;
                                     50 AD      B0      AD  9E 01020 217$:    MOVAB   INTMED_DATA, R0                         ; 2615
                                                      1946  31 01024          BRW     551$                                    ;
                                80000000 8F           B0  AD  D1 01027 218$:    CMPL    INTMED_DATA, #-2147483648               ; 2620
                                                        04  12 0102F          BNEQ    219$                                    ;
                                     23 AD      FF      AD  E8 01031          BLBS    SRC_INFO+7, 221$                        ;
                                        AD      B0      AD  D5 01035 219$:    TSTL    INTMED_DATA                             ; 2625
                                                        15  18 01038          BGEQ    220$                                    ;
                                           00000000G    00  DD 0103A          PUSHL   DBG$GL_OPCODE_NAME                      ;
                                                        01  DD 01040          PUSHL   #1                                      ;
                                           000286A3     8F  DD 01042          PUSHL   #165539                                 ;
                                00000000G  00          03  FB 01048          CALLS   #3, LIB$SIGNAL                          ;
                                        05      FF      AD  E9 0104F 220$:    BLBC    SRC_INFO+7, 221$                        ; 2626
                                     B0 AD      B0      AD  CE 01053          MNEGL   INTMED_DATA, INTMED_DATA                ;
                                     34 BE      B0      AD  D0 01058 221$:    MOVL    INTMED_DATA, @OUTPUT                    ; 2627
                                                        19  11 0105D 222$:    BRB     226$                                    ; 2588
                                     52 F5      AD      3C  0105F 223$:    MOVZWL  DST_INFO+5, R2                          ; 2637
                                     50 01      CE      01  01063          MNEGL   #1, I                                   ; 2639
                                                        0C  11 01066          BRB     225$                                    ;
                                51 B0 AD           01      50  EF 01068 224$:    EXTZV   I, #1, INTMED_DATA, R1                  ;
                             34 BE    01           50      51  F0 0106E          INSV    R1, I, #1, @OUTPUT                      ;
                                      F0           50      52  F2 01074 225$:    AOBLSS  R2, I, 224$                             ; 2637
                                                      1F5D  31 01078 226$:    BRW     649$                                    ; 2187
                                     58 AE                32  B0 0107B 227$:    MOVW    #50, CLASS_S_DESC                       ; 2650
                                     5C AE           60   AE  9E 0107F          MOVAB   TEMP_BUF2, CLASS_S_DESC+4               ; 2651
                                     0B                   56  D1 01084          CMPL    R6, #11                                 ; 2655
                                                        23  12 01087          BNEQ    229$                                    ;
                                     51      FF7C        CD  9E 01089          MOVAB   TEMP_BUF1, R1                           ; 2657
                                     50      B0          AD  9E 0108E          MOVAB   INTMED_DATA, R0                         ;
                                        00000000G        00  16 01092          JSB     DBG$CVT_CVTROUH_R1                      ;
                                     06      FF          AD  E9 01098          BLBC    SRC_INFO+7, 228$                        ; 2658
                                FF7D CD      80          8F  88 0109C          BISB2   #128, TEMP_BUF1+1                       ;
                                                        01  DD 010A2 228$:    PUSHL   #1                                      ; 2659
                                                        7E  7C 010A4          CLRQ    -(SP)                                   ;
                                     3C AE              DD  010A6             PUSHL   SCALE                                   ;
                                           00C4         31  010A9             BRW     239$                                    ;
                                     11                 56  D1 010AC 229$:    CMPL    R6, #17                                 ; 2662
                                                        21  12 010AF          BNEQ    231$                                    ;
                                     B1 AD              95  010B1             TSTB    INTMED_DATA+1                           ; 2664
                                                        04  18 010B4          BGEQ    230$                                    ;
                                FF   AD                 01  88 010B6          BISB2   #1, SRC_INFO+7                          ;
                                                        01  DD 010BA 230$:    PUSHL   #1                                      ; 2665
                                                        7E  7C 010BC          CLRQ    -(SP)                                   ;
                                     3C AE              DD  010BE             PUSHL   SCALE                                   ;
                                                        7E  D4 010C1          CLRL    -(SP)                                   ;
                                     6C AE              9F  010C3             PUSHAB  CLASS_S_DESC                            ;
                                     B0 AD              9F  010C6             PUSHAB  INTMED_DATA                             ;
                                00000000G  00           07  FB 010C9          CALLS   #7, FOR$CVT_D_TF                        ;
                                                        30  11 010D0          BRB     234$                                    ;
                                     17                 56  D1 010D2 231$:    CMPL    R6, #23                                 ; 2668
                                                        3E  12 010D5          BNEQ    236$                                    ;
                                     B1 AD              95  010D7             TSTB    INTMED_DATA+1                           ; 2670
                                                        04  18 010DA          BGEQ    232$                                    ;
                                FF   AD                 01  88 010DC          BISB2   #1, SRC_INFO+7                          ;
                                     1B      04         BE  91 010E0 232$:    CMPB    @4(SP), #27                             ; 2671
```

```
                                 06  13 010E4                BEQL    233$                                                    ;
                          1D     04  BE 91 010E6                CMPB    @4(SP), #29                                     2672    ;
                                     18 12 010EA                BNEQ    235$                                            ;
                                     01 DD 010EC 233$:          PUSHL   #1                                              ;
                                     7E 7C 010EE                CLRQ    -(SP)                                           ;
                                  3C AE DD 010F0                PUSHL   SCALE                                           ;
                                     7E D4 010F3                CLRL    -(SP)                                           ;
                                  6C AE 9F 010F5                PUSHAB  CLASS_S_DESC                                    ;
                                  B0 AD 9F 010F8                PUSHAB  INTMED_DATA                                     ;
                00000000G   00       07 FB 010FB                CALLS   #7, FOR$CVT_G_TF                                ;
                                     7C 11 01102 234$:          BRB     241$                                            ;
                                     01 DD 01104 235$:          PUSHL   #1                                              ;
                                     7E 7C 01106                CLRQ    -(SP)                                           ;
                                  3C AE DD 01108                PUSHL   SCALE                                           ;
                                     7E D4 0110B                CLRL    -(SP)                                           ;
                                  6C AE 9F 0110D                PUSHAB  CLASS_S_DESC                                    ;
                                  B0 AD 9F 01110                PUSHAB  INTMED_DATA                                     ;
                                     64 11 01113                BRB     240$                                            ;
                          23         56 D1 01115 236$:          CMPL    R6, #35                                 2679    ;
                                     69 12 01118                BNEQ    242$                                            ;
                       58 AE     FD  AD B0 0111A                MOVW    SRC_INFO+5, CLASS_S_DESC                2681    ;
                       5C AE     F9  AD D0 0111F                MOVL    SRC_INFO+1, CLASS_S_DESC+4              2682    ;
                          7E     55  8F 9A 01124                MOVZBL  #85, -(SP)                              2684    ;
                          7E     34  AE CE 01128                MNEGL   SCALE, -(SP)                            2683    ;
                                     7E D4 0112C                CLRL    -(SP)                                           ;
                                FF7C CD 9F 0112E                PUSHAB  TEMP_BUF1                                       ;
                                  68 AE 9F 01132                PUSHAB  CLASS_S_DESC                                    ;
                00000000G   00       05 FB 01135                CALLS   #5, OTS$CVT_T_H                                 ;
                            6E       50 D0 01138                MOVL    R0, STATUS                                      ;
                            15       6E E8 0113F                BLBS    STATUS, 237$                            2685    ;
                                00000000G 00 DD 01142           PUSHL   DBG$GL_OPCODE_NAME                              ;
                                          01 DD 01148           PUSHL   #1                                              ;
                                00028298  8F DD 0114A           PUSHL   #164504                                         ;
                00000000G   00       03 FB 01150                CALLS   #3, LIB$SIGNAL                                  ;
                                FF7D CD 95 01157 237$:          TSTB    TEMP_BUF1+1                             2686    ;
                                     04 18 0115B                BGEQ    238$                                            ;
                             FF  AD      01 88 0115D            BISB2   #1, SRC_INFO+7                                  ;
                             58  AE      32 B0 01161 238$:      MOVW    #50, CLASS_S_DESC                       2687    ;
                             5C  AE   60  AE 9E 01165           MOVAB   TEMP_BUF2, CLASS_S_DESC+4               2688    ;
                                          01 DD 0116A           PUSHL   #1                                      2689    ;
                                          7E 7C 0116C           CLRQ    -(SP)                                           ;
                                          7E D4 0116E           CLRL    -(SP)                                           ;
                                          7E D4 01170 239$:     CLRL    -(SP)                                           ;
                                       6C AE 9F 01172           PUSHAB  CLASS_S_DESC                                    ;
                                     FF7C CD 9F 01175           PUSHAB  TEMP_BUF1                                       ;
                00000000G   00       07 FB 01179 240$:          CALLS   #7, FOR$CVT_H_TF                                ;
                            6E       50 D0 01180 241$:          MOVL    R0, STATUS                                      ;
                            15       6E E8 01183 242$:          BLBS    STATUS, 243$                            2693    ;
                                00000000G 00 DD 01186           PUSHL   DBG$GL_OPCODE_NAME                              ;
                                          01 DD 0118C           PUSHL   #1                                              ;
                                00028A3A  8F DD 0118E           PUSHL   #166458                                         ;
                00000000G   00       03 FB 01194                CALLS   #3, LIB$SIGNAL                                  ;
                60 AE               32       20 3B 01198 243$:  SKPC    #32, #50, TEMP_BUF2                     2694    ;
                                             02 12 011A0        BNEQ    244$                                            ;
                                             51 D4 011A2        CLRL    R1                                              ;
                                50      60   AE 9E 011A4 244$:  MOVAB   TEMP_BUF2, R0                                  ;
                             5A             51    50 C3 011A8  SUBL3   R0, R1, BUF_OFFSET                              ;
```

```
                2C   AE          30            5A  C3 011AC           SUBL3   BUF_OFFSET, #48, NO_DIGITS                ; 2695
                  06             0F            6B  8F 011B1           CASEB   (R1T), #15, #6                            ; 2697
          0130          00DA          009E          0046     011B5 245$:  .WORD   252$-245$,-
                        01D3          018C          018C     011BD               255$-245$,-
                                                                                259$-245$,-
                                                                                264$-245$,-
                                                                                270$-245$,-
                                                                                270$-245$,-
                                                                                274$-245$
                                      0B                      56  D1 011C3           CMPL    R6, #11                   ; 2768
                                                              08  12 011C6           BNEQ    246$
                                          00000000'           EF  9F 011C8           PUSHAB  P.AGT                     ; 2769
                                                              29  11 011CE           BRB     251$
                                      11                      56  D1 011D0 246$:     CMPL    R6, #17               ; 2770
                                                              08  12 011D3           BNEQ    247$
                                          00000000'           EF  9F 011D5           PUSHAB  P.AGU                     ; 2771
                                                              1C  11 011DB           BRB     251$
                                      17                      56  D1 011DD 247$:     CMPL    R6, #23               ; 2772
                                                              09  12 011E0           BNEQ    249$
                                          00000000'           EF  9F 011E2           PUSHAB  P.AGV                     ; 2773
                                                        1DDE  31  11 011E8 248$:     BRW     647$
                                      23                      56  D1 011EB 249$:     CMPL    R6, #35               ; 2774
                                                              03  13 011EE           BEQL    250$
                                                        1DE5  31  11 011F0           BRW     649$
                                          00000000'           EF  9F 011F3 250$:     PUSHAB  P.AGW                     ; 2775
                                                              ED  11 011F9 251$:     BRB     248$
                                      15        FF            AD  E9 011FB 252$:     BLBC    SRC_INFO+7, 253$          ; 2702
                                          00000000G           00  DD 011FF           PUSHL   DBG$GL_OPCODE_NAME
                                                              01  DD 01205           PUSHL   #1
                                          00028EF0            8F  DD 01207           PUSHL   #167664
                                00000000G   00                    03  FB 0120D           CALLS   #3, LIB$SIGNAL
                                            57                    69  3C 01214 253$:     MOVZWL  (R9), R7              ; 2703
                                            57            2C      AE  D1 01217           CMPL    NO_DIGITS, R7
                                                                  15  15 0121B           BLEQ    254$
                                              00000000G           00  DD 0121D           PUSHL   DBG$GL_OPCODE_NAME
                                                                  01  DD 01223           PUSHL   #1
                                              00028A3A            8F  DD 01225           PUSHL   #166458
                                00000000G   00                    03  FB 0122B           CALLS   #3, LIB$SIGNAL
                          50                57            2C      AE  C3 01232 254$:     SUBL3   NO_DIGITS, R7, R0     ; 2704
          50              30                6E                    00  2C 01237           MOVC5   #0, (SP), #48, R0, TEMP_BUF1
                                                          FF7C      CD     0123C
                                            50            FF7C  CD47  9E 0123F           MOVAB   TEMP_BUF1[R7], R0     ; 2706
                                            50              2C      AE  C2 01245           SUBL2   NO_DIGITS, R0
                          60          61  AE4A              2C      AE  28 01249           MOVC3   NO_DIGITS, TEMP_BUF2+1[BUF_OFFSET], (R0)
                                                                00E5  31  11 01250           BRW     268$              ; 2707
                                                                      69  B5 01253 255$:     TSTW    (R9)              ; 2716
                                                                      04  12 01255           BNEQ    256$
                                                                      54  D4 01257           CLRL    DES_LEN
                                                                      05  11 01259           BRB     257$
                                            54                        69  3C 0125B 256$:     MOVZWL  (R9), DES_LEN
                                                                      54  D7 0125E           DECL    DES_LEN
                                          2C    AE                    54  D1 01260 257$:     CMPL    DES_LEN, NO_DIGITS    ; 2718
                                                                      15  18 01264           BGEQ    258$
                                              00000000G               00  DD 01266           PUSHL   DBG$GL_OPCODE_NAME
                                                                      01  DD 0126C           PUSHL   #1
                                              00028A3A                8F  DD 0126E           PUSHL   #166458
                                00000000G   00                        03  FB 01274           CALLS   #3, LIB$SIGNAL
```

```
 FF7C   CD        54       60 AE4A        2C   AE  09 0127B 258$:   CVTSP   NO_DIGITS, TEMP_BUF2[BUF_OFFSET], DES_LEN, -; 2719
                                                                            TEMP_BUF1                                 ;
   34   BE        54       FF7C   CD           54  08 01285         CVTPS   DES_LEN, TEMP_BUF1, DES_LEN, @OUTPUT      ; 2720
                                               54  11 0128D         BRB     263$                                      ; 2697
                                  52      01   AA  9E 0128F 259$:   MOVAB   1(R10), R2                                ; 2725
        52        30              6E           00  2C 01293         MOVC5   #0, (SP), #48, R2, TEMP_BUF2              ;
                                          60   AE     01298                                                           ;
   2C   AE        69              10           00  ED 0129A         CMPZV   #0, #16, (R9), NO_DIGITS                  ; 2726
                                               15  18 012A0         BGEQ    260$                                      ;
                                  000000000G   00  DD 012A2         PUSHL   DBG$GL_OPCODE_NAME                        ;
                                               01  DD 012A8         PUSHL   #1                                        ;
                                  00028A3A     8F  DD 012AA         PUSHL   #166458                                   ;
                     000000000G   00           03  FB 012B0         CALLS   #3, LIB$SIGNAL                            ;
                                  5A           69  3C 012B7 260$:   MOVZWL  (R9), BUF_OFFSET                          ; 2727
                  5A              31           5A  C3 012BA         SUBL3   BUF_OFFSET, #49, BUF_OFFSET               ;
                                  51      60 AE4A  9E 012BE         MOVAB   TEMP_BUF2[BUF_OFFSET], R1                 ; 2728
                                  50           61  9A 012C3         MOVZBL  (R1), R0                                  ;
                                  50 000000000G0040  9E 012C6       MOVAB   LIB$AB_CVT_U_O-48[R0], R0                 ; 2729
                                  06      FF   AD  E9 012CE         BLBC    SRC_INFO+7, 261$                          ; 2728
                                  50      0A   A0  D0 012D2         MOVL    10(R0), R0                                ;
                                               03  11 012D6         BRB     262$                                      ;
                                  50           60  D0 012D8 261$:   MOVL    (R0), R0                                  ; 2729
                                  61           50  90 012DB 262$:   MOVB    R0, (R1)                                  ; 2728
             34   BE              61           69  28 012DE         MOVC3   (R9), (R1), @OUTPUT                       ; 2731
                                               5A  11 012E3 263$:   BRB     269$                                      ; 2697
                                               69  B5 012E5 264$:   TSTW    (R9)                                      ; 2740
                                               04  12 012E7         BNEQ    265$                                      ;
                                               57  D4 012E9         CLRL    DES_LEN                                   ;
                                               05  11 012EB         BRB     266$                                      ;
                                  57           69  3C 012ED 265$:   MOVZWL  (R9), DES_LEN                             ;
                                               57  D7 012F0         DECL    DES_LEN                                   ;
                                  57      2C   AE  D1 012F2 266$:   CMPL    NO_DIGITS, DES_LEN                        ; 2742
                                               15  15 012F6         BLEQ    267$                                      ;
                                  000000000G   00  DD 012F8         PUSHL   DBG$GL_OPCODE_NAME                        ;
                                               01  DD 012FE         PUSHL   #1                                        ;
                                  00028A3A     8F  DD 01300         PUSHL   #166458                                   ;
                     000000000G   00           03  FB 01306         CALLS   #3, LIB$SIGNAL                            ;
                  50              57      2C   AE  C3 0130D 267$:   SUBL3   NO_DIGITS, DES_LEN, R0                    ; 2743
                                               50  D6 01312         INCL    R0                                        ;
        50        30              6E           00  2C 01314         MOVC5   #0, (SP), #48, R0, TEMP_BUF1              ;
                                       FF7C    CD     01319                                                           ;
                       10   AE         FF7C CD47   9E 0131C         MOVAB   TEMP_BUF1[DES_LEN], 16(SP)                ; 2744
                  50   10   AE            2C   AE  C3 01323         SUBL3   NO_DIGITS, 16(SP), R0                     ;
                  60   61 AE4A            2C   AE  28 01329         MOVC3   NO_DIGITS, TEMP_BUF2+1[BUF_OFFSET], (R0)  ;
                       10   BE         60 AE4A     90 01330         MOVB    TEMP_BUF2[BUF_OFFSET], @16(SP)            ; 2745
                                               57  D6 01336         INCL    R7                                        ; 2747
             34   BE     FF7C   CD             57  28 01338 268$:   MOVC3   R7, TEMP_BUF1, @OUTPUT                    ;
                                               6B  11 0133F 269$:   BRB     276$                                      ; 2697
   2C   AE        69              10           00  ED 01341 270$:   CMPZV   #0, #16, (R9), NO_DIGITS                  ; 2752
                                               15  18 01347         BGEQ    271$                                      ;
                                  000000000G   00  DD 01349         PUSHL   DBG$GL_OPCODE_NAME                        ;
                                               01  DD 0134F         PUSHL   #1                                        ;
                                  00028A3A     8F  DD 01351         PUSHL   #166458                                   ;
                     000000000G   00           03  FB 01357         CALLS   #3, LIB$SIGNAL                            ;
 FF7C   CD        69       60 AE4A        2C   AE  09 0135E 271$:   CVTSP   NO_DIGITS, TEMP_BUF2[BUF_OFFSET], (R9), -   ; 2753
                                                                            TEMP_BUF1                                 ;
                                  13           6B  91 01368         CMPB    (R11), #19                                ; 2755
```

```
                                                09  12 0136B                BNEQ    272$                                    ;
                                     50 00000000G  00  9E 0136D                MOVAB   LIB$AB_CVTPT_0, R0                      ;
                                                07  11 01374                BRB     273$                                    ;
                                     50 00000000G  00  9E 01376 272$:         MOVAB   LIB$AB_CVTPT_2, R0                      ;
            69          60      FF7C    CD          69  24 0137D 273$:         CVTPT   (R9), TEMP_BUF1, (R0), (R9), @OUTPUT    ; 2756
                                                34  BE 01384                                                                 ;
                                                24  11 01386                BRB     276$                                    ; 2697
                                     1F         2C  AE  D1 01388 274$:         CMPL    NO_DIGITS, #31                          ; 2761
                                                15  15 0138C                BLEQ    275$                                    ;
                                         00000000G  00  DD 0138E                PUSHL   DBG$GL_OPCODE_NAME                      ;
                                                01  DD 01394                PUSHL   #1                                      ;
                                         00028A3A   8F  DD 01396                PUSHL   #166458                                 ;
                                 00000000G      00      03  FB 0139C            CALLS   #3, LIB$SIGNAL                          ;
    34  BE          69          60  AE4A        2C  AE  09 013A3 275$:         CVTSP   NO_DIGITS, TEMP_BUF2[BUF_OFFSET], (R9), - ; 2762
                                                                                        @OUTPUT                                 ;
                                                1C29  31 013AC 276$:         BRW     649$                                    ; 2187
                                     02             6B  91 013AF 277$:         CMPB    (R11), #2                               ; 2784
                                                12  13 013B2                BEQL    280$                                    ;
                                     0E             6B  91 013B4               CMPB    (R11), #14                              ;
                                                0D  13 013B7                BEQL    280$                                    ;
                                     25             6B  91 013B9               CMPB    (R11), #37                              ;
                                                03  1E 013BC                BGEQU   279$                                    ;
                                                035C  31 013BE 278$:         BRW     327$                                    ;
                                     27             6B  91 013C1 279$:         CMPB    (R11), #39                              ;
                                                F8  1A 013C4                BGTRU   278$                                    ;
                                 58  AE             32  B0 013C6 280$:         MOVW    #50, CLASS_S_DESC                       ; 2786
                                 5C  AE         60  AE  9E 013CA               MOVAB   TEMP_BUF2, CLASS_S_DESC+4               ; 2787
                                                57  D4 013CF                CLRL    DIGITS_IN_FRACT                         ; 2800
                                                52  D4 013D1                CLRL    R2                                      ; 2801
                                                58  D5 013D3                TSTL    BIN_SCALE                               ;
                                                1C  13 013D5                BEQL    281$                                    ;
                                                52  D6 013D7                INCL    R2                                      ;
                                            30  AE  D5 013D9                TSTL    SCALE                                   ;
                                                15  13 013DC                BEQL    281$                                    ;
                                     00000000'  EF  9F 013DE                PUSHAB  P.AGX                                   ; 2803
                                                01  DD 013E4                PUSHL   #1                                      ;
                                         00028362   8F  DD 013E6                PUSHL   #164706                                 ;
                                 00000000G      00      03  FB 013EC            CALLS   #3, LIB$SIGNAL                          ;
                                                58  D5 013F3 281$:         TSTL    BIN_SCALE                               ; 2804
                                                03  18 013F5                BGEQ    282$                                    ;
                                     57         58  CE 013F7                MNEGL   BIN_SCALE, DIGITS_IN_FRACT              ; 2806
                                            30  AE  D5 013FA 282$:         TSTL    SCALE                                   ; 2807
                                                04  18 013FD                BGEQ    283$                                    ;
                                     57     30  AE  CE 013FF                MNEGL   SCALE, DIGITS_IN_FRACT                  ; 2809
                                     06             56  D1 01403 283$:         CMPL    R6, #6                                  ; 2814
                                                4E  12 01406                BNEQ    287$                                    ;
                                 FF7C   CD      B0  AD  6E 01408               CVTLD   INTMED_DATA, TEMP_BUF1                  ; 2816
                                                58  D5 0140E 284$:         TSTL    BIN_SCALE                               ; 2821
                                                16  18 01410                BGEQ    285$                                    ;
                                     51     FF7C   CD  9E 01412               MOVAB   TEMP_BUF1, R1                           ; 2823
                                     50 00000000'  EF  9E 01417               MOVAB   P.AGY, R0                               ;
                                         00000000G  00  16 0141E                JSB     DBG$CVT_DIVD2_R1                        ;
                                                58  D6 01424                INCL    BIN_SCALE                               ; 2824
                                                E6  11 01426                BRB     284$                                    ; 2821
                                                16  15 01428 285$:         BLEQ    286$                                    ; 2826
                                     51     FF7C   CD  9E 0142A               MOVAB   TEMP_BUF1, R1                           ; 2828
                                     50 00000000'  EF  9E 0142F               MOVAB   P.AGZ, R0                               ;
```

```
                        00000000G  00  16 01436          JSB     DBG$CVT_MULD2_R1
                                   58  D7 0143C          DECL    BIN_SCALE                       ;2829
                                   E8  11 0143E          BRB     285$                            ;2826
                                30 AE  DD 01440 286$:    PUSHL   SCALE                           ;2832
                                   57  DD 01443          PUSHL   DIGITS_IN_FRACT
                                60 AE  9F 01445          PUSHAB  CLASS_S_DESC
                             FF7C CD   9F 01448          PUSHAB  TEMP_BUF1
                00000000G    00        04 FB 0144C          CALLS   #4, FOR$CVT_D_TF
                                   015A 31 01453          BRW     303$
                             0C        56 D1 01456 287$:    CMPL    R6, #12                         ;2835
                                   03  13 01459          BEQL    288$
                                   00FC 31 0145B          BRW     300$
                             1A    04  BE 91 0145E 288$:    CMPB    @4(SP), #26                     ;2836
                                   4E  13 01462          BEQL    292$
                             51 FF7C   CD 9E 01464          MOVAB   TEMP_BUF1, R1                   ;2839
                             50 B0     AD 9E 01469          MOVAB   INTMED_DATA, R0
                                00000000G  00 16 0146D          JSB     DBG$CVT_CVTROUH_R1
                             06    FF  AD E9 01473          BLBC    SRC_INFO+7, 289$                ;2840
                          FF7D CD  80  8F 88 01477          BISB2   #128, TEMP_BUF1+1
                                   58  D5 0147D 289$:    TSTL    BIN_SCALE                       ;2846
                                   16  18 0147F          BGEQ    290$
                             51 FF7C   CD 9E 01481          MOVAB   TEMP_BUF1, R1                   ;2848
                             50 00000000'  EF 9E 01486          MOVAB   P.AHA, R0
                                00000000G  00 16 0148D          JSB     DBG$CVT_DIVH2_R1
                                   58  D6 01493          INCL    BIN_SCALE                       ;2849
                                   E6  11 01495          BRB     289$                            ;2846
                                   03  14 01497 290$:    BGTR    291$                            ;2851
                                   0101 31 01499          BRW     302$
                             51 FF7C   CD 9E 0149C 291$:    MOVAB   TEMP_BUF1, R1                   ;2853
                             50 00000000'  EF 9E 014A1          MOVAB   P.AH8, R0
                                00000000G  00 16 014A8          JSB     DBG$CVT_MULH2_R1
                                   58  D7 014AE          DECL    BIN_SCALE                       ;2854
                                   E5  11 014B0          BRB     290$                            ;2851
                             15    52  E9 014B2 292$:    BLBC    R2, 293$                        ;2869
                                00000000'  EF 9F 014B5          PUSHAB  P.AHC                   ;2871
                                   01  DD 014B8          PUSHL   #1
                                00028362   8F DD 014BD          PUSHL   #164706
                00000000G    00        03 FB 014C3          CALLS   #3, LIB$SIGNAL
                                58 AE  B4 014CA 293$:    CLRW    CLASS_S_DESC                    ;2873
48 AE           B0 AD                  10 28 014CD          MOVC3   #16, INTMED_DATA, PREVIOUS_VALUE ;2880
48 AE           B0 AD                  10 28 014D3 294$:    MOVC3   #16, INTMED_DATA, PREVIOUS_VALUE ;2900
                   50  B0 AD 9E 014D9          MOVAB   INTMED_DATA, R0                         ;2905
                00000000G  00 16 014DD          JSB     DBG$CVT_SCALE_OU_DOWN_BY_10_R1
38 AE           B0 AD                  10 28 014E3          MOVC3   #16, INTMED_DATA, SAVED_VALUE   ;2912
                   50  B0 AD 9E 014E9          MOVAB   INTMED_DATA, R0                         ;2917
                00000000G  00 16 014ED          JSB     DBG$CVT_SCALE_OU_UP_BY_10_R1
                   51  58 AE  3C 014F3          MOVZWL  CLASS_S_DESC, CURRENT_POSITION          ;2925
                   09  11 014F7          BRB     296$
50              51  5C AE  C1 014F9 295$:    ADDL3   CLASS_S_DESC+4, CURRENT_POSITION, R0
                01 A0                  60 90 014FE          MOVB    (R0), 1(R0)
                   F4                  51 F4 01502 296$:    SOBGEQ  CURRENT_POSITION, 295$
50              48 AE  B0 AD           C3 01505          SUBL3   INTMED_DATA, PREVIOUS_VALUE, R0 ;2930
5C BE              50                  30 81 01508          ADDB3   #48, R0, @CLASS_S_DESC+4
                   58  AE B6 01510          INCW    CLASS_S_DESC                            ;2937
B0 AD           38 AE                  10 28 01513          MOVC3   #16, SAVED_VALUE, INTMED_DATA   ;2944
                   BC  AD D5 01519          TSTL    INTMED_DATA+12                          ;2947
                       B5  12 0151C          BNEQ    294$
```

```
                                                B8  AD  D5 0151E          TSTL    INTMED_DATA+8                                   ; 2948
                                                B0  12  01521            BNEQ    294$
                                                B4  AD  D5 01523          TSTL    INTMED_DATA+4                                   ; 2949
                                                AB  12  01526            BNEQ    294$
                                                B0  AD  D5 01528          TSTL    INTMED_DATA                                     ; 2950
                                                A6  12  0152B            BNEQ    294$
                                        19      FF  AD  E9 0152D          BLBC    SRC_INFO+7, 299$                                ; 2955
                                        51      58  AE  3C 01531          MOVZWL  CLASS_S_DESC, CURRENT_POSITION                  ; 2961
                                                09  11  01535          BRB     298$
                                50      51      5C  AE  C1 01537 297$:   ADDL3   CLASS_S_DESC+4, CURRENT_POSITION, R0
                                    01  A0                  60  90 0153C          MOVB    (R0), 1(R0)
                                        F4                  51  F4 01540 298$:   SOBGEQ  CURRENT_POSITION, 297$
                                    5C  BE                  2D  90 01543          MOVB    #45, @CLASS_S_DESC+4                    ; 2963
                                                58  AE  B6 01547          INCW    CLASS_S_DESC                                    ; 2964
                                        50      58  AE  3C 0154A 299$:   MOVZWL  CLASS_S_DESC, R0                                ; 2971
                                        50      5C  AE  C0 0154E          ADDL2   CLASS_S_DESC+4, R0
                                        60          2E  90 01552          MOVB    #46, (R0)
                                        6E          01  D0 01555          MOVL    #1, STATUS                                      ; 2973
                                                59  11  01558          BRB     304$                                            ; 2836
                                        1E          56  D1 0155A 300$:   CMPL    R6, #30                                         ; 2976
                                                54  12  0155D          BNEQ    304$
                                        15          52  E9 0155F          BLBC    R2, 301$                                        ; 2981
                                    00000000'       EF  9F 01562          PUSHAB  P.AHD                                           ; 2983
                                                01  DD  01568          PUSHL   #1
                                    00028362        8F  DD 0156A          PUSHL   #164706
                            00000000G   00          03  FB 01570          CALLS   #3, LIB$SIGNAL
                                    2C  AE      FD  AD  3C 01577 301$:   MOVZWL  SRC_INFO+5, NO_DIGITS                           ; 2985
60  AE      2C  AE      B0  AD      2C  AE      08  0157C          CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -                    ; 2986
                                                                            TEMP_BUF2
                58  AE      2C  AE          01  A1  01585          ADDW3   #1, NO_DIGITS, CLASS_S_DESC                             ; 2987
                                                15  DD  0158B          PUSHL   #21                                             ; 2989
                                                7E  7C  0158D          CLRQ    -(SP)                                           ; 2988
                                        FF7C    CD  9F  0158F          PUSHAB  TEMP_BUF1
                                            68  AE  9F  01593          PUSHAB  CLASS_S_DESC
                            00000000G   00          05  FB 01596          CALLS   #5, OTS$CVT_T_H
                                            30  AE  DD  0159D 302$:   PUSHL   SCALE                                           ; 2990
                                                57  DD  015A0          PUSHL   DIGITS_IN_FRACT
                                            60  AE  9F  015A2          PUSHAB  CLASS_S_DESC
                                        FF7C    CD  9F  015A5          PUSHAB  TEMP_BUF1
                            00000000G   00          04  FB 015A9          CALLS   #4, FOR$CVT_H_TF
                                        6E          50  D0 015B0 303$:   MOVL    R0, STATUS
60  AE                      32                      20  3B 015B3 304$:   SKPC    #32, #50, TEMP_BUF2                             ; 2994
                                                02  12  015B8          BNEQ    305$
                                                51  D4  015BA          CLRL    R1
                                        50      60  AE  9E  015BC 305$:   MOVAB   TEMP_BUF2, R0
                                    5A      51          50  C3 015C0          SUBL3   R0, R1, BUF_OFFSET
                                    52      32          5A  C3 015C4          SUBL3   BUF_OFFSET, #50, R2                             ; 2995
60  AE4A                        52                      20  3A 015C8          LOCC    #32, R2, TEMP_BUF2[BUF_OFFSET]
                                                02  12  015CE          BNEQ    306$
                                                51  D4  015D0          CLRL    R1
                                                51  D5  015D2 306$:   TSTL    NEXT_BLANK                                      ; 2996
                                                05  12  015D4          BNEQ    307$
                                        55      52  D0  015D6          MOVL    R2, FINAL_LEN                                   ; 2998
                                                0B  11  015D9          BRB     308$
                                        51      5A  C2  015DB 307$:   SUBL2   BUF_OFFSET, R1                                  ; 3000
                                        50      60  AE  9E  015DE          MOVAB   TEMP_BUF2, R0
                                    55      51          50  C3 015E2          SUBL3   R0, R1, FINAL_LEN
```

```
                                                57  D5 015E6 308$:   TSTL    DIGITS_IN_FRACT                         ; 3001
                                                02  12 015E8         BNEQ    309$                                    ;
                                                55  D7 015EA         DECL    FINAL_LEN                               ; 3003
                                            72  6E  E8 015EC 309$:   BLBS    STATUS, 317$                            ; 3005
                                        58  AE  1E  32 B0 015EF         MOVW    #50, CLASS_S_DESC                       ; 3008
                                                56  D1 015F3         CMPL    R6, #30                                 ; 3009
                                                05  12 015F6         BNEQ    310$                                    ;
                                            57  1F  D0 015F8         MOVL    #31, DIGITS_IN_FRACT                    ; 3011
                                                1D  11 015FB         BRB     313$                                    ;
                                            09  F5  AD  B1 015FD 310$:   CMPW    DST_INFO+5, #9                          ; 3013
                                                05  1A 01601         BGTRU   311$                                    ;
                                            57  21  D0 01603         MOVL    #33, DIGITS_IN_FRACT                    ; 3015
                                                12  11 01606         BRB     313$                                    ;
                                            50  F5  AD  3C 01608 311$:   MOVZWL  DST_INFO+5, R0                         ; 3017
                                            50  09  C2 0160C         SUBL2   #9, R0                                  ;
                                            21  50  D1 0160F         CMPL    R0, #33                                 ;
                                                03  15 01612         BLEQ    312$                                    ;
                                            50  21  D0 01614         MOVL    #33, R0                                 ;
                                            57  50  D0 01617 312$:   MOVL    R0, DIGITS_IN_FRACT                     ;
                                                04  DD 0161A 313$:   PUSHL   #4                                      ; 3018
                                                7E  D4 0161C         CLRL    -(SP)                                   ;
                                                38  AE  DD 0161E         PUSHL   SCALE                                   ;
                                                57  DD 01621         PUSHL   DIGITS_IN_FRACT                         ;
                                                68  AE  9F 01623         PUSHAB  CLASS_S_DESC                            ;
                                            FF7C  CD  9F 01626         PUSHAB  TEMP_BUF1                               ;
                                    00000000G   00  06  FB 0162A         CALLS   #6, FOR$CVT_H_TE                        ;
                                                6E  50  D0 01631         MOVL    R0, STATUS                              ;
                                                15  6E  E8 01634         BLBS    STATUS, 314$                            ; 3019
                                                    00000000'   EF  9F 01637         PUSHAB  P.AHE                           ;
                                                    01  DD 0163D         PUSHL   #1                                      ;
                                                    00028362    8F  DD 0163F         PUSHL   #164706                         ;
                                                00000000G   00  03  FB 01645         CALLS   #3, LIB$SIGNAL                  ;
                                        60  AE      32  20  3B 0164C 314$:   SKPC    #32, #50, TEMP_BUF2                     ; 3020
                                                    02  12 01651         BNEQ    316$                                    ;
                                                    51  D4 01653 315$:   CLRL    R1                                      ;
                                                50      60  AE  9E 01655 316$:   MOVAB   TEMP_BUF2, R0                      ;
                                            5A      51  50  C3 01659         SUBL3   R0, R1, BUF_OFFSET                      ;
                                            55      32  5A  C3 0165D         SUBL3   BUF_OFFSET, #50, FINAL_LEN              ; 3021
                                                14  AE  55  D0 01661 317$:   MOVL    FINAL_LEN, OUTPUT_STR_LEN               ; 3024
                                                26  6B  91 01665         CMPB    (R11), #38                              ; 3027
                                                    46  12 01668         BNEQ    321$                                    ;
                                                58  AE  55  B0 0166A         MOVW    FINAL_LEN, CLASS_S_DESC                 ; 3031
                                        5C  AE  34  AE  01  C1 0166E         ADDL3   #1, OUTPUT, CLASS_S_DESC+4              ; 3032
                                                52      58  AE  9E 01674         MOVAB   CLASS_S_DESC, R2                        ; 3033
                                                51      60  AE4A  9E 01678         MOVAB   TEMP_BUF2[BUF_OFFSET], R1       ;
                                                50  55  D0 0167D         MOVL    FINAL_LEN, R0                           ;
                                                    00000000G   00  16 01680         JSB     LIB$SCOPY_R_DX6                 ;
                                                6E  50  D0 01686         MOVL    R0, STATUS                              ;
                                            00000000G   8F  6E  D1 01689         CMPL    STATUS, #LIB$_STRTRU                    ; 3034
                                                15  12 01690         BNEQ    319$                                    ;
                                                    00000000G   00  DD 01692 318$:   PUSHL   DBG$GL_OPCODE_NAME              ;
                                                    01  DD 01698         PUSHL   #1                                      ;
                                                    000286AB    8F  DD 0169A         PUSHL   #165547                         ;
                                            00000000G   00  03  FB 016A0         CALLS   #3, LIB$SIGNAL                  ;
                                                03  6E  E9 016A7 319$:   BLBC    STATUS, 320$                            ; 3035
                                                    1761  31 016AA         BRW     622$                                    ;
                                                    1755  31 016AD 320$:   BRW     621$                                    ;
```

```
                27            6B  91 016B0 321$:   CMPB    (R11), #39                      ; 3039
                              45  12 016B3         BNEQ    325$
              58  AE            55  B0 016B5         MOVW    FINAL_LEN, CLASS_S_DESC         ; 3043
              5C  AE        34  AE  D0 016B9         MOVL    OUTPUT, CLASS_S_DESC+4          ; 3044
                  52        58  AE  9E 016BE         MOVAB   CLASS_S_DESC, R2                ; 3045
                  51        60 AE4A 9E 016C2         MOVAB   TEMP_BUF2[BUF_OFFSET], R1
                  50            55  D0 016C7         MOVL    FINAL_LEN, R0
                     00000000G  00  16 016CA         JSB     LIB$SCOPY_R_DX6
                  6E            50  D0 016D0         MOVL    R0, STATUS
     00000000G    8F            6E  D1 016D3         CMPL    STATUS, #LIB$_STRTRU            ; 3046
                                15  12 016DA         BNEQ    323$
                     00000000G  00  DD 016DC 322$:    PUSHL   DBG$GL_OPCODE_NAME
                                01  DD 016E2         PUSHL   #1
                     000286AB   8F  DD 016E4         PUSHL   #165547
     00000000G      00            03  FB 016EA         CALLS   #3, LIB$SIGNAL
                  03            6E  E9 016F1 323$:    BLBC    STATUS, 324$                    ; 3047
                              176D  31 016F4         BRW     627$
                              1761  31 016F7 324$:    BRW     626$
                  51        60 AE4A 9E 016FA 325$:    MOVAB   TEMP_BUF2[BUF_OFFSET], R1       ; 3053
                  52            59  D0 016FF         MOVL    R9, R2
                  50            55  D0 01702         MOVL    FINAL_LEN, R0
                     00000000G  00  16 01705         JSB     LIB$SCOPY_R_DX6
                  6E            50  D0 0170B         MOVL    R0, STATUS
     00000000G    8F            6E  D1 0170E         CMPL    STATUS, #LIB$_STRTRU            ; 3054
                                03  13 01715         BEQL    326$
                              1840  31 01717         BRW     641$
                              1828  31 0171A 326$:    BRW     640$
                06              56  D1 0171D 327$:    CMPL    R6, #6                          ; 3063
                                08  12 01720         BNEQ    328$
                     00000000'  EF  9F 01722         PUSHAB  P.AHF                           ; 3064
                                1B  11 01728         BRB     331$
              0C                56  D1 0172A 328$:    CMPL    R6, #12                         ; 3065
                                08  12 0172D         BNEQ    329$
                     00000000'  EF  9F 0172F         PUSHAB  P.AHG                           ; 3066
                                0E  11 01735         BRB     331$
              1E                56  D1 01737 329$:    CMPL    R6, #30                         ; 3067
                                03  13 0173A         BEQL    330$
                             1899  31 0173C         BRW     649$
                     00000000'  EF  9F 0173F 330$:    PUSHAB  P.AHH                           ; 3068
                             1881  31 01745 331$:    BRW     647$
                                58  D5 01748 332$:    TSTL    BIN_SCALE                       ; 3074
                                26  15 0174A         BLEQ    333$
              51        B0        AD  9E 0174C         MOVAB   INTMED_DATA, R1
              50 00000000'        EF  9E 01750         MOVAB   P.AHI, R0
                 00000000G        00  16 01757         JSB     DBG$CVT_MULD2_R1
              51        B8        AD  9E 0175D         MOVAB   INTMED_DATA+8, R1
              50 00000000'        EF  9E 01761         MOVAB   P.AHJ, R0
                 00000000G        00  16 01768         JSB     DBG$CVT_MULD2_R1
                                58  D7 0176E         DECL    BIN_SCALE
                                D6  11 01770         BRB     332$
                                26  18 01772 333$:    BGEQ    334$
              51        B0        AD  9E 01774         MOVAB   INTMED_DATA, R1
              50 00000000'        EF  9E 01778         MOVAB   P.AHK, R0
                 00000000G        00  16 0177F         JSB     DBG$CVT_DIVD2_R1
              51        B8        AD  9E 01785         MOVAB   INTMED_DATA+8, R1
              50 00000000'        EF  9E 01789         MOVAB   P.AHL, R0
                 00000000G        00  16 01790         JSB     DBG$CVT_DIVD2_R1
```

```
                                  58  D6 01796          INCL    BIN_SCALE                                  ;
                                  D8  11 01798          BRB     333$                                       ;
                            30    AE  D5 0179A 334$:    TSTL    SCALE                                      ;
                                  27  15 0179D          BLEQ    335$                                       ;
                     51     B0    AD  9E 0179F          MOVAB   INTMED_DATA, R1                            ;
                     50 00000000' EF  9E 017A3          MOVAB   P.AHM, R0                                  ;
                        00000000G 00  16 017AA          JSB     DBG$CVT_MULD2_R1                           ;
                     51     B8    AD  9E 017B0          MOVAB   INTMED_DATA+8, R1                          ;
                     50 00000000' EF  9E 017B4          MOVAB   P.AHN, R0                                  ;
                        00000000G 00  16 017BB          JSB     DBG$CVT_MULD2_R1                           ;
                            30    AE  D7 017C1          DECL    SCALE                                      ;
                                  D4  11 017C4          BRB     334$                                       ;
                                  27  18 017C6 335$:    BGEQ    336$                                       ;
                     51     B0    AD  9E 017C8          MOVAB   INTMED_DATA, R1                            ;
                     50 00000000' EF  9E 017CC          MOVAB   P.AHO, R0                                  ;
                        00000000G 00  16 017D3          JSB     DBG$CVT_DIVD2_R1                           ;
                     51     B8    AD  9E 017D9          MOVAB   INTMED_DATA+8, R1                          ;
                     50 00000000' EF  9E 017DD          MOVAB   P.AHP, R0                                  ;
                        00000000G 00  16 017E4          JSB     DBG$CVT_DIVD2_R1                           ;
                            30    AE  D6 017EA          INCL    SCALE                                      ;
                                  D7  11 017ED          BRB     335$                                       ;
                     08     0C    AE  E9 017EF 336$:    BLBC    CVT_ROUND_FLAG, 337$                       ; 3076
              FF7C   CD     B0    AD  6B 017F3          CVTRDL  INTMED_DATA, TEMP_BUF1                     ; 3078
                                  06  11 017F9          BRB     338$                                       ;
              FF7C   CD     B0    AD  6A 017FB 337$:    CVTDL   INTMED_DATA, TEMP_BUF1                     ; 3080
          29                01    6B  8F 01801 338$:    CASEB   (R11), #1, #41                             ; 3082
0054      046F      044C      005D       01805 339$:    .WORD   341$-339$,-                                ;
0D92      0CD3      0C77      0054       0180D                  383$-339$,-                                ;
0054      0054      0054      0054       01815                  385$-339$,-                                ;
0054      0054      0054      0054       0181D                  340$-339$,-                                ;
0054      0054      0054      0054       01825                  479$-339$,-                                ;
0054      0054      0054      0054       0182D                  486$-339$,-                                ;
0054      0054      0054      0054       01835                  498$-339$,-                                ;
0054      0054      0054      0054       0183D                  340$-339$,-                                ;
0054      0054      005D      0054       01845                  340$-339$,-                                ;
005D      0054      0054      0054       0184D                  340$-339$,-                                ;
                    005D      005D       01855                  340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
                                                                340$-339$,-                                ;
```

```
                                                                                340$-339$,-
                                                                                340$-339$,-
                                                                                341$-339$,-
                                                                                340$-339$,-
                                                                                340$-339$,-
                                                                                340$-339$,-
                                                                                340$-339$,-
                                                                                340$-339$,-
                                                                                341$-339$,-
                                                                                341$-339$,-
                                                                                341$-339$
                                00000000'   EF   9F 01859 340$:   PUSHAB  P.AHQ                                    3123
                                      1767  31   0185F             BRW     647$
                                  52   F5   AD   3C 01862 341$:   MOVZWL  DST_INFO+5, R2                           3116
                                  50        01   CE 01866         MNEGL   #1, I                                    3118
                                            0C   11 01869         BRB     343$
      51        B0   AD             01        50   EF 0186B 342$:   EXTZV   I, #1, INTMED_DATA, R1
34    BE             01             50        51   F0 01871         INSV    R1, I, #1, @OUTPUT
                     F0             50        52   F2 01877 343$:   AOBLSS  R2, I, 342$                            3116
                                           175A   31 0187B         BRW     649$                                     2187
                                              58  D5 0187E 344$:   TSTL    BIN_SCALE                                3128
                                              26  15 01880         BLEQ    345$
                                  51        B0   AD   9E 01882         MOVAB   INTMED_DATA, R1
                                  50 00000000'   EF   9E 01886         MOVAB   P.AHR, R0
                                     00000000G   00   16 0188D         JSB     DBG$CVT_MULD2_R1
                                  51        B8   AD   9E 01893         MOVAB   INTMED_DATA+8, R1
                                  50 00000000'   EF   9E 01897         MOVAB   P.AHS, R0
                                     00000000G   00   16 0189E         JSB     DBG$CVT_MULD2_R1
                                              58  D7 018A4         DECL    BIN_SCALE
                                              D6  11 018A6         BRB     344$
                                              26  18 018A8 345$:   BGEQ    346$
                                  51        B0   AD   9E 018AA         MOVAB   INTMED_DATA, R1
                                  50 00000000'   EF   9E 018AE         MOVAB   P.AHT, R0
                                     00000000G   00   16 018B5         JSB     DBG$CVT_DIVD2_R1
                                  51        B8   AD   9E 018BB         MOVAB   INTMED_DATA+8, R1
                                  50 00000000'   EF   9E 018BF         MOVAB   P.AHU, R0
                                     00000000G   00   16 018C6         JSB     DBG$CVT_DIVD2_R1
                                              58  D6 018CC         INCL    BIN_SCALE
                                              D8  11 018CE         BRB     345$
                                        30    AE   D5 018D0 346$:   TSTL    SCALE
                                              27  15 018D3         BLEQ    347$
                                  51        B0   AD   9E 018D5         MOVAB   INTMED_DATA, R1
                                  50 00000000'   EF   9E 018D9         MOVAB   P.AHV, R0
                                     00000000G   00   16 018E0         JSB     DBG$CVT_MULD2_R1
                                  51        B8   AD   9E 018E6         MOVAB   INTMED_DATA+8, R1
                                  50 00000000'   EF   9E 018EA         MOVAB   P.AHW, R0
                                     00000000G   00   16 018F1         JSB     DBG$CVT_MULD2_R1
                                        30    AE   D7 018F7         DECL    SCALE
                                              D4  11 018FA         BRB     346$
                                              27  18 018FC 347$:   BGEQ    348$
                                  51        B0   AD   9E 018FE         MOVAB   INTMED_DATA, R1
                                  50 00000000'   EF   9E 01902         MOVAB   P.AHX, R0
                                     00000000G   00   16 01909         JSB     DBG$CVT_DIVD2_R1
                                  51        B8   AD   9E 0190F         MOVAB   INTMED_DATA+8, R1
                                  50 00000000'   EF   9E 01913         MOVAB   P.AHY, R0
                                     00000000G   00   16 0191A         JSB     DBG$CVT_DIVD2_R1
                                        30    AE   D6 01920         INCL    SCALE
```

```
                                          D7  11 01923        BRB     347$                          ;
                16            04          6B  8F 01925 348$:  CASEB   (R11), #4, #22                ; 3130
  002E          002E          0072      0037     01929 349$:  .WORD   351$-349$,-                   ;
  002E          002E          0072      002E     01931                355$-349$,-                   ;
  002E          002E          002E      002E     01939                350$-349$,-                   ;
  002E          002E          002E      002E     01941                350$-349$,-                   ;
  002E          002E          002E      002E     01949                350$-349$,-                   ;
                0083          002E      002E     01951                355$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      350$-349$,-                   ;
                                                                      356$-349$                     ;
                              00000000'   EF  9F 01957 350$:  PUSHAB  P.AHZ                         ; 3156
                                        1669  31 0195D        BRW     647$                          ;
                        10    BE      B0  AD  71 01960 351$:  CMPD    INTMED_DATA, @LRGST_D_LU      ; 3135
                                          50  DC 01965        MOVPSL  R0                            ;
      50            50        02          02  EF 01967        EXTZV   #2, #2, R0, R0                ;
                              15          50  F4 0196C        SOBGEQ  R0, 352$                      ;
                              00000000G   00  DD 0196F        PUSHL   DBG$GL_OPCODE_NAME            ;
                                          01  DD 01975        PUSHL   #1                            ;
                              000286A3    8F  DD 01977        PUSHL   #165539                       ;
                   00000000G  00          03  FB 0197D        CALLS   #3, LIB$SIGNAL                ;
                                    00E0  8F  B9 01984 352$:  BICPSW  #224                          ; 3136
                              07      0C  AE  E9 01988        BLBC    CVT_ROUND_FLAG, 353$          ; 3137
                        34    BE      B0  AD  6B 0198C        CVTRDL  INTMED_DATA, @OUTPUT          ; 3139
                                          05  11 01991        BRB     354$                          ;
                        34    BE      B0  AD  6A 01993 353$:  CVTDL   INTMED_DATA, @OUTPUT          ; 3141
                                        11A6  31 01998 354$:  BRW     583$                          ; 3143
                              50      B0  AD  9E 0199B 355$:  MOVAB   INTMED_DATA, R0               ; 3147
                              51      34  AE  D0 0199F        MOVL    OUTPUT, R1                    ;
                              00000000G   00  16 019A3        JSB     DBG$CVT_CVTRDQ_R1             ;
                                        162C  31 019A9        BRW     649$                          ;
                              51    FF7C  CD  9E 019AC 356$:  MOVAB   TEMP_BUF1, R1                 ; 3151
                              50      B0  AD  9E 019B1        MOVAB   INTMED_DATA, R0               ;
                              00000000G   00  16 019B5        JSB     DBG$CVT_CVTDH_R1              ;
                              50    FF7C  CD  9E 019BB        MOVAB   TEMP_BUF1, R0                 ; 3152
                                        0501  31 019C0        BRW     409$                          ;
                              02          6B  91 019C3 357$:  CMPB    (R11), #2                     ; 3164
                                          0F  13 019C6        BEQL    358$                          ;
                              0E          6B  91 019C8        CMPB    (R11), #14                    ;
                                          0A  13 019CB        BEQL    358$                          ;
                              25          6B  91 019CD        CMPB    (R11), #37                    ;
                                          76  1F 019D0        BLSSU   367$                          ;
                              27          6B  91 019D2        CMPB    (R11), #39                    ;
```

```
                                                71  1A 019D5                BGTRU   367$
                                        58      AE 32  B0 019D7 358$:       MOVW    #50, CLASS_S_DESC                                       ;3166
                                        5C      AE 60  AE 9E 019DB          MOVAB   TEMP_BUF2, CLASS_S_DESC+4                               ;3167
                                    01      0A     04  BE BF 019E0          CASEB   @4(SP), #10, #1                                         ;3170
                                          0009         0004 019E5 359$:      .WORD   360$-359$,-
                                                                                    361$-359$
                                          57             07 D0 019E9 360$:     MOVL    #7, DIGITS_IN_FRACT
                                                         03 11 019EC           BRB     362$
                                          57             10 D0 019EE 361$:     MOVL    #16, DIGITS_IN_FRACT
                                          07         F5  AD B1 019F1 362$:     CMPW    DST_INFO+5, #7                                  ;3180
                                                         15 1B 019F5           BLEQU   364$
                                          51         F5  AD 3C 019F7           MOVZWL  DST_INFO+5, R1                                  ;3183
                                          51             07 C2 019FB           SUBL2   #7, R1
                                          50             57 D0 019FE           MOVL    DIGITS_IN_FRACT, R0
                                          51             50 D1 01A01           CMPL    R0, R1
                                                         03 15 01A04           BLEQ    363$
                                          50             51 D0 01A06           MOVL    R1, R0
                                          57             50 D0 01A09 363$:     MOVL    R0, DIGITS_IN_FRACT                             ;3182
                                                        7E D4 01A0C 364$:      CLRL    -(SP)                                           ;3184
                                                    34  AE DD 01A0E           PUSHL   SCALE
                                                    57  DD 01A11             PUSHL   DIGITS_IN_FRACT
                                                    64  AE 9F 01A13           PUSHAB  CLASS_S_DESC
                                                    B0  AD 9F 01A16           PUSHAB  INTMED_DATA
                                  00000000G         00     05  FB 01A19       CALLS   #5, FOR$CVT_D_TE
                                                    6E     50  D0 01A20       MOVL    R0, STATUS
                                                    15     6E  E8 01A23       BLBS    STATUS, 365$                                    ;3185
                                              00000000'    EF  9F 01A26       PUSHAB  P.AIA
                                                           01  DD 01A2C       PUSHL   #1
                                              00028362     8F  DD 01A2E       PUSHL   #164706
                                      00000000G            00  FB 01A34       CALLS   #3, LIB$SIGNAL
                                60    AE                   32  20 3B 01A3B 365$:  SKPC    #32, #50, TEMP_BUF2                             ;3186
                                                                03 13 01A40       BEQL    366$
                                                              FC10 31 01A42       BRW     316$
                                                              FC0B 31 01A45 366$: BRW     315$
                                              00000000'        EF 9F 01A48 367$: PUSHAB  P.AIB                                           ;3226
                                                             1578 31 01A4E       BRW     647$
                                                      1B         04  BE 91 01A51 368$:  CMPB    @4(SP), #27                             ;3231
                                                                 09  13 01A55       BEQL    369$
                                                      1D         04  BE 91 01A57       CMPB    @4(SP), #29                             ;3232
                                                                 03  13 01A5B       BEQL    369$
                                                               00CB  31 01A5D       BRW     374$
                                                      51       FF7C  CD 9E 01A60 369$:  MOVAB   TEMP_BUF1, R1                           ;3233
                                                      50         B0  AD 9E 01A65       MOVAB   INTMED_DATA, R0
                                                  00000000G      00  16 01A69       JSB     DBG$CVT_CVTGH_R1
                                                      51         C0  AD 9E 01A6F       MOVAB   INTMED_DATA+16, R1
                                                      50         B8  AD 9E 01A73       MOVAB   INTMED_DATA+8, R0
                                                  00000000G      00  16 01A77       JSB     DBG$CVT_CVTGH_R1
                                B0    AD          FF7C   CD             10  28 01A7D       MOVC3   #16, TEMP_BUF1, INTMED_DATA
                                                                       58  D5 01A84 370$:  TSTL    BIN_SCALE
                                                                       26  15 01A86       BLEQ    371$
                                                      51         B0  AD 9E 01A88       MOVAB   INTMED_DATA, R1
                                                      50  00000000'  EF 9E 01A8C       MOVAB   P.AIC, R0
                                                  00000000G      00  16 01A93       JSB     DBG$CVT_MULH2_R1
                                                      51         C0  AD 9E 01A99       MOVAB   INTMED_DATA+16, R1
                                                      50  00000000'  EF 9E 01A9D       MOVAB   P.AID, R0
                                                  00000000G      00  16 01AA4       JSB     DBG$CVT_MULH2_R1
                                                                     58  D7 01AAA       DECL    BIN_SCALE
```

```
                    D6  11 01AAC          BRB     370$                                    ;
                    26  18 01AAE 371$:     BGEQ    372$                                    ;
51          B0      AD  9E 01AB0          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01AB4          MOVAB   P.AIE, R0                               ;
   00000000G        00  16 01ABB          JSB     DBG$CVT_DIVH2_R1                        ;
51          C0      AD  9E 01AC1          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01AC5          MOVAB   P.AIF, R0                               ;
   00000000G        00  16 01ACC          JSB     DBG$CVT_DIVH2_R1                        ;
                    58  D6 01AD2          INCL    BIN_SCALE                               ;
                    D8  11 01AD4          BRB     371$                                    ;
            30      AE  D5 01AD6 372$:     TSTL    SCALE                                   ;
                    27  15 01AD9          BLEQ    373$                                    ;
51          B0      AD  9E 01ADB          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01ADF          MOVAB   P.AIG, R0                               ;
   00000000G        00  16 01AE6          JSB     DBG$CVT_MULH2_R1                        ;
51          C0      AD  9E 01AEC          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01AF0          MOVAB   P.AIH, R0                               ;
   00000000G        00  16 01AF7          JSB     DBG$CVT_MULH2_R1                        ;
            30      AE  D7 01AFD          DECL    SCALE                                   ;
                    D4  11 01B00          BRB     372$                                    ;
                    51  18 01B02 373$:     BGEQ    375$                                    ;
51          B0      AD  9E 01B04          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01B08          MOVAB   P.AII, R0                               ;
   00000000G        00  16 01B0F          JSB     DBG$CVT_DIVH2_R1                        ;
51          C0      AD  9E 01B15          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01B19          MOVAB   P.AIJ, R0                               ;
   00000000G        00  16 01B20          JSB     DBG$CVT_DIVH2_R1                        ;
            30      AE  D6 01B26          INCL    SCALE                                   ;
                    D7  11 01B29          BRB     373$                                    ;
                    58  D5 01B2B 374$:     TSTL    BIN_SCALE                               ; 3235
                    26  15 01B2D          BLEQ    375$                                    ;
51          B0      AD  9E 01B2F          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01B33          MOVAB   P.AIK, R0                               ;
   00000000G        00  16 01B3A          JSB     DBG$CVT_MULH2_R1                        ;
51          C0      AD  9E 01B40          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01B44          MOVAB   P.AIL, R0                               ;
   00000000G        00  16 01B4B          JSB     DBG$CVT_MULH2_R1                        ;
                    58  D7 01B51          DECL    BIN_SCALE                               ;
                    D6  11 01B53          BRB     374$                                    ;
                    58  D5 01B55 375$:     TSTL    BIN_SCALE                               ;
                    26  18 01B57          BGEQ    376$                                    ;
51          B0      AD  9E 01B59          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01B5D          MOVAB   P.AIM, R0                               ;
   00000000G        00  16 01B64          JSB     DBG$CVT_DIVH2_R1                        ;
51          C0      AD  9E 01B6A          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01B6E          MOVAB   P.AIN, R0                               ;
   00000000G        00  16 01B75          JSB     DBG$CVT_DIVH2_R1                        ;
                    58  D6 01B7B          INCL    BIN_SCALE                               ;
                    D6  11 01B7D          BRB     375$                                    ;
            30      AE  D5 01B7F 376$:     TSTL    SCALE                                   ;
                    27  15 01B82          BLEQ    377$                                    ;
51          B0      AD  9E 01B84          MOVAB   INTMED_DATA, R1                         ;
50 00000000'        EF  9E 01B88          MOVAB   P.AIO, R0                               ;
   00000000G        00  16 01B8F          JSB     DBG$CVT_MULH2_R1                        ;
51          C0      AD  9E 01B95          MOVAB   INTMED_DATA+16, R1                      ;
50 00000000'        EF  9E 01B99          MOVAB   P.AIP, R0                               ;
   00000000G        00  16 01BA0          JSB     DBG$CVT_MULH2_R1                        ;
```

```
                              30   AE   D7 01BA6        DECL    SCALE                          ;
                                   D4   11 01BA9        BRB     376$                           ;
                                   27   18 01BAB 377$:  BGEQ    378$                           ;
                    51        B0   AD   9E 01BAD        MOVAB   INTMED_DATA, R1                ;
                    50 00000000'   EF   9E 01BB1        MOVAB   P.AIQ, R0                      ;
                       00000000G   00   16 01BB8        JSB     DBG$CVT_DIVH2_R1               ;
                    51        C0   AD   9E 01BBE        MOVAB   INTMED_DATA+16, R1             ;
                    50 00000000'   EF   9E 01BC2        MOVAB   P.AIR, R0                      ;
                       00000000G   00   16 01BC9        JSB     DBG$CVT_DIVH2_R1               ;
                              30   AE   D6 01BCF        INCL    SCALE                          ;
                                   D7   11 01BD2        BRB     377$                           ;
                    11        0C   AE   E9 01BD4 378$:  BLBC    CVT_ROUND_FLAG, 379$           ; 3237
                    51      FF7C   CD   9E 01BD8        MOVAB   TEMP_BUF1, R1                  ; 3239
                    50        B0   AD   9E 01BDD        MOVAB   INTMED_DATA, R0                ;
                       00000000G   00   16 01BE1        JSB     DBG$CVT_CVTRHL_R1              ;
                                   07   11 01BE7        BRB     380$                           ;
              FF7C   CD       B0   AD6AFD  01BE9 379$:  CVTHL   INTMED_DATA, TEMP_BUF1         ; 3241
                 29   01           6B   8F 01BF0 380$:  CASEB   (R11), #1, #41                 ; 3242
   0054          0080          005D      00A3  01BF4 381$:  .WORD   387$-381$,-                    ;
   09A3          08E4          0888      0054  01BFC                383$-381$,-                    ;
   0054          0054          0054      0054  01C04                385$-381$,-                    ;
   0054          0054          0054      0054  01C0C                382$-381$,-                    ;
   0054          0054          0054      0054  01C14                382$-381$,-                    ;
   0054          0054          0054      0054  01C1C                479$-381$,-                    ;
   0054          0054          0054      0054  01C24                486$-381$,-                    ;
   0054          0054          0054      0054  01C2C                498$-381$,-                    ;
   0054          0054          00A3      0054  01C34                382$-381$,-                    ;
   00A3          0054          0054      0054  01C3C                382$-381$,-                    ;
                               00A3      00A3  01C44                382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    387$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
                                                                    382$-381$,-                    ;
```

```
                                                                      387$-381$,-      ;
                                                                      387$-381$,-      ;
                                                                      387$-381$        ;
                                 00000000'  EF  9F 01C48 382$:  PUSHAB P.AIS            ; 3283
                                     1378   31 01C4E           BRW    647$             ;
                   000000FF  8F    FF7C  CD  D1 01C51 383$:  CMPL   TEMP_BUF1, #255  ; 3247
                                       15   1B 01C5A           BLEQU  384$             ;
                                 00000000G  00  DD 01C5C           PUSHL  DBG$GL_OPCODE_NAME ;
                                       01   DD 01C62           PUSHL  #1               ;
                                 000286A3  8F  DD 01C64           PUSHL  #165539          ;
                   00000000G  00             03  FB 01C6A           CALLS  #3, LIB$SIGNAL   ;
                                     0CEA   31 01C71 384$:  BRW    549$             ; 3248
                   0000FFFF  8F    FF7C  CD  D1 01C74 385$:  CMPL   TEMP_BUF1, #65535 ; 3253
                                       15   1B 01C7D           BLEQU  386$             ;
                                 00000000G  00  DD 01C7F           PUSHL  DBG$GL_OPCODE_NAME ;
                                       01   DD 01C85           PUSHL  #1               ;
                                 000286A3  8F  DD 01C87           PUSHL  #165539          ;
                   00000000G  00             03  FB 01C8D           CALLS  #3, LIB$SIGNAL   ;
                                     0D3C   31 01C94 386$:  BRW    560$             ; 3254
                             52      F5  AD  3C 01C97 387$:  MOVZWL DST_INFO+5, R2   ; 3276
                             50          01  CE 01C9B           MNEGL  #1, I            ; 3278
                                         0C  11 01C9E           BRB    389$             ;
       51        B0  AD      01          50  EF 01CA0 388$:  EXTZV  I, #1, INTMED_DATA, R1 ;
   34  BE            01      50          51  F0 01CA6           INSV   R1, I, #1, @OUTPUT ;
                     F0      50          52  F2 01CAC 389$:  AOBLSS R2, I, 388$       ; 3276
                                     1325   31 01CB0           BRW    649$             ; 2187
                             1B      04  BE  91 01CB3 390$:  CMPB   @4(SP), #27      ; 3289
                                         09  13 01CB7           BEQL   391$             ;
                             1D      04  BE  91 01CB9           CMPB   @4(SP), #29      ; 3290
                                         03  13 01CBD           BEQL   391$             ;
                                     00CB   31 01CBF           BRW    396$             ;
                             51    FF7C  CD  9E 01CC2 391$:  MOVAB  TEMP_BUF1, R1    ; 3291
                             50      B0  AD  9E 01CC7           MOVAB  INTMED_DATA, R0  ;
                                 00000000G  00  16 01CCB           JSB    DBG$CVT_CVTGH_R1 ;
                             51      C0  AD  9E 01CD1           MOVAB  INTMED_DATA+16, R1 ;
                             50      B8  AD  9E 01CD5           MOVAB  INTMED_DATA+8, R0 ;
                                 00000000G  00  16 01CD9           JSB    DBG$CVT_CVTGH_R1 ;
                 B0  AD    FF7C  CD      10  28 01CDF           MOVC3  #16, TEMP_BUF1, INTMED_DATA ;
                                         58  D5 01CE6 392$:  TSTL   BIN_SCALE        ;
                                         26  15 01CE8           BLEQ   393$             ;
                             51      B0  AD  9E 01CEA           MOVAB  INTMED_DATA, R1  ;
                             50 00000000'  EF  9E 01CEE           MOVAB  P.AIT, R0        ;
                                 00000000G  00  16 01CF5           JSB    DBG$CVT_MULH2_R1 ;
                             51      C0  AD  9E 01CFB           MOVAB  INTMED_DATA+16, R1 ;
                             50 00000000'  EF  9E 01CFF           MOVAB  P.AIU, R0        ;
                                 00000000G  00  16 01D06           JSB    DBG$CVT_MULH2_R1 ;
                                         58  D7 01D0C           DECL   BIN_SCALE        ;
                                         D6  11 01D0E           BRB    392$             ;
                                         26  18 01D10 393$:  BGEQ   394$             ;
                             51      B0  AD  9E 01D12           MOVAB  INTMED_DATA, R1  ;
                             50 00000000'  EF  9E 01D16           MOVAB  P.AIV, R0        ;
                                 00000000G  00  16 01D1D           JSB    DBG$CVT_DIVH2_R1 ;
                             51      C0  AD  9E 01D23           MOVAB  INTMED_DATA+16, R1 ;
                             50 00000000'  EF  9E 01D27           MOVAB  P.AIW, R0        ;
                                 00000000G  00  16 01D2E           JSB    DBG$CVT_DIVH2_R1 ;
                                         58  D6 01D34           INCL   BIN_SCALE        ;
                                         D8  11 01D36           BRB    393$             ;
```

```
                30 AE D5 01D38 394$:   TSTL    SCALE                                                                   ;
                27 15 01D3B            BLEQ    395$                                                                    ;
          51     B0 AD 9E 01D3D        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01D41     MOVAB   P.AIX, R0                                                               ;
             00000000G 00 16 01D48     JSB     DBG$CVT_MULH2_R1                                                        ;
          51     C0 AD 9E 01D4E        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01D52     MOVAB   P.AIY, R0                                                               ;
             00000000G 00 16 01D59     JSB     DBG$CVT_MULH2_R1                                                        ;
                30 AE D7 01D5F         DECL    SCALE                                                                   ;
                D4 11 01D62            BRB     394$                                                                    ;
                51 18 01D64 395$:      BGEQ    397$                                                                    ;
          51     B0 AD 9E 01D66        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01D6A     MOVAB   P.AIZ, R0                                                               ;
             00000000G 00 16 01D71     JSB     DBG$CVT_DIVH2_R1                                                        ;
          51     C0 AD 9E 01D77        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01D7B     MOVAB   P.AJA, R0                                                               ;
             00000000G 00 16 01D82     JSB     DBG$CVT_DIVH2_R1                                                        ;
                30 AE D6 01D88         INCL    SCALE                                                                   ;
                D7 11 01D8B            BRB     395$                                                                    ;
                58 D5 01D8D 396$:      TSTL    BIN_SCALE                                                               ; 3293
                26 15 01D8F            BLEQ    397$                                                                    ;
          51     B0 AD 9E 01D91        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01D95     MOVAB   P.AJB, R0                                                               ;
             00000000G 00 16 01D9C     JSB     DBG$CVT_MULH2_R1                                                        ;
          51     C0 AD 9E 01DA2        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01DA6     MOVAB   P.AJC, R0                                                               ;
             00000000G 00 16 01DAD     JSB     DBG$CVT_MULH2_R1                                                        ;
                58 D7 01DB3            DECL    BIN_SCALE                                                               ;
                D6 11 01DB5            BRB     396$                                                                    ;
                58 D5 01DB7 397$:      TSTL    BIN_SCALE                                                               ;
                26 18 01DB9            BGEQ    398$                                                                    ;
          51     B0 AD 9E 01DBB        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01DBF     MOVAB   P.AJD, R0                                                               ;
             00000000G 00 16 01DC6     JSB     DBG$CVT_DIVH2_R1                                                        ;
          51     C0 AD 9E 01DCC        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01DD0     MOVAB   P.AJE, R0                                                               ;
             00000000G 00 16 01DD7     JSB     DBG$CVT_DIVH2_R1                                                        ;
                58 D6 01DDD            INCL    BIN_SCALE                                                               ;
                D6 11 01DDF            BRB     397$                                                                    ;
                30 AE D5 01DE1 398$:   TSTL    SCALE                                                                   ;
                27 15 01DE4            BLEQ    399$                                                                    ;
          51     B0 AD 9E 01DE6        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01DEA     MOVAB   P.AJF, R0                                                               ;
             00000000G 00 16 01DF1     JSB     DBG$CVT_MULH2_R1                                                        ;
          51     C0 AD 9E 01DF7        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01DFB     MOVAB   P.AJG, R0                                                               ;
             00000000G 00 16 01E02     JSB     DBG$CVT_MULH2_R1                                                        ;
                30 AE D7 01E08         DECL    SCALE                                                                   ;
                D4 11 01E0B            BRB     398$                                                                    ;
                27 18 01E0D 399$:      BGEQ    400$                                                                    ;
          51     B0 AD 9E 01E0F        MOVAB   INTMED_DATA, R1                                                         ;
          50 00000000' EF 9E 01E13     MOVAB   P.AJH, R0                                                               ;
             00000000G 00 16 01E1A     JSB     DBG$CVT_DIVH2_R1                                                        ;
          51     C0 AD 9E 01E20        MOVAB   INTMED_DATA+16, R1                                                      ;
          50 00000000' EF 9E 01E24     MOVAB   P.AJI, R0                                                               ;
             00000000G 00 16 01E2B     JSB     DBG$CVT_DIVH2_R1                                                        ;
                30 AE D6 01E31         INCL    SCALE                                                                   ;
```

```
                                                D7  11 01E34          BRB     399$
                                    16      04  6B  8F 01E36 400$:   CASEB   (R11), #4, #22                          ;3295
                                002E    002E    007F    0037 01E3A 401$:   .WORD   403$-401$,-
                                002E    002E    007F    002E 01E42         407$-401$,-
                                002E    002E    002E    002E 01E4A         402$-401$,-
                                002E    002E    002E    002E 01E52         402$-401$,-
                                002E    002E    002E    002E 01E5A         402$-401$,-
                                002E    0086    002E    002E 01E62         407$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           402$-401$,-
                                                                           408$-401$
                                00000000'   EF  9F 01E68 402$:   PUSHAB  P.AJJ                                      ;3317
                                      1158  31  31 01E6E         BRW     647$
                                50      B0  AD  9E 01E71 403$:   MOVAB   INTMED_DATA, R0                            ;3300
                                51      18  AE  D0 01E75         MOVL    LRGST_A_LU, R1
                                00000000G   00  16 01E79         JSB     DBG$CVT_CMPH_R1
                                            50  D5 01E7F         TSTL    R0
                                            15  15 01E81         BLEQ    404$
                                00000000G   00  DD 01E83         PUSHL   DBG$GL_OPCODE_NAME
                                            01  DD 01E89         PUSHL   #1
                                000286A3    8F  DD 01E8B         PUSHL   #165539
                00000000G   00              03  FB 01E91         CALLS   #3, LIB$SIGNAL
                                    00E0    8F  B9 01E98 404$:   BICPSW  #224                                        ;3301
                                10      0C  AE  E9 01E9C         BLBC    CVT_ROUND_FLAG, 405$                     ;3302
                                50      B0  AD  9E 01EA0         MOVAB   INTMED_DATA, R0                          ;3304
                                51      34  AE  D0 01EA4         MOVL    OUTPUT, R1
                                00000000G   00  16 01EA8         JSB     DBG$CVT_CVTRHL_R1
                                            06  11 01EAE         BRB     406$
                            34  BE      B0  AD6AFD 01EB0 405$:   CVTHL   INTMED_DATA, @OUTPUT                      ;3306
                                      0C88  31  31 01EB6 406$:   BRW     583$                                     ;3307
                                50      B0  AD  9E 01EB9 407$:   MOVAB   INTMED_DATA, R0                          ;3311
                                      0CC9  31  31 01EBD         BRW     586$
                                50      B0  AD  9E 01EC0 408$:   MOVAB   INTMED_DATA, R0                          ;3314
                                51      34  AE  D0 01EC4 409$:   MOVL    OUTPUT, R1
                                      0866  31  31 01EC8         BRW     518$
                                1B      04  BE  91 01ECB 410$:   CMPB    @4(SP), #27                              ;3323
                                            09  13 01ECF         BEQL    411$
                                1D      04  BE  91 01ED1         CMPB    @4(SP), #29                              ;3324
                                            03  13 01ED5         BEQL    411$
                                      00CB  31  31 01ED7         BRW     416$
                                51    FF7C  CD  9E 01EDA 411$:   MOVAB   TEMP_BUF1, R1                           ;3325
                                50      B0  AD  9E 01EDF         MOVAB   INTMED_DATA, R0
                                00000000G   00  16 01EE3         JSB     DBG$CVT_CVTGH_R1
```

```
                                51        C0   AD  9E 01EE9          MOVAB   INTMED_DATA+16, R1                                ;
                                50        B8   AD  9E 01EED          MOVAB   INTMED_DATA+8, R0                                ;
                            00000000G     00   16  01EF1          JSB     DBG$CVT_CVTGH_R1                                 ;
                    B0  AD      FF7C    CD                 10  28  01EF7          MOVC3   #16, TEMP_BUFT, INTMED_DATA              ;
                                                           58  D5  01EFE 412$:     TSTL    BIN_SCALE                                ;
                                                           26  15  01F00          BLEQ    413$                                     ;
                                51        B0   AD  9E 01F02          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01F06          MOVAB   P.AJK, R0                                        ;
                                    00000000G     00  16 01F0D          JSB     DBG$CVT_MULH2_R1                                 ;
                                51        C0   AD  9E 01F13          MOVAB   INTMED_DATA+16, R1                               ;
                                50  00000000'    EF  9E 01F17          MOVAB   P.AJL, R0                                        ;
                                    00000000G     00  16 01F1E          JSB     DBG$CVT_MULH2_R1                                 ;
                                                  58  D7 01F24          DECL    BIN_SCALE                                        ;
                                                  D6  11 01F26          BRB     412$                                             ;
                                                  26  18 01F28 413$:     BGEQ    414$                                             ;
                                51        B0   AD  9E 01F2A          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01F2E          MOVAB   P.AJM, R0                                        ;
                                    00000000G     00  16 01F35          JSB     DBG$CVT_DIVH2_R1                                 ;
                                51        C0   AD  9E 01F3B          MOVAB   INTMED_DATA+16, R1                               ;
                                50  00000000'    EF  9E 01F3F          MOVAB   P.AJN, R0                                        ;
                                    00000000G     00  16 01F46          JSB     DBG$CVT_DIVH2_R1                                 ;
                                                  58  D6 01F4C          INCL    BIN_SCALE                                        ;
                                                  D8  11 01F4E          BRB     413$                                             ;
                                        30        AE  D5 01F50 414$:     TSTL    SCALE                                            ;
                                                  27  15 01F53          BLEQ    415$                                             ;
                                51        B0   AD  9E 01F55          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01F59          MOVAB   P.AJO, R0                                        ;
                                    00000000G     00  16 01F60          JSB     DBG$CVT_MULH2_R1                                 ;
                                51        C0   AD  9E 01F66          MOVAB   INTMED_DATA+16, R1                               ;
                                50  00000000'    EF  9E 01F6A          MOVAB   P.AJP, R0                                        ;
                                    00000000G     00  16 01F71          JSB     DBG$CVT_MULH2_R1                                 ;
                                        30        AE  D7 01F77          DECL    SCALE                                            ;
                                                  D4  11 01F7A          BRB     414$                                             ;
                                                  51  18 01F7C 415$:     BGEQ    417$                                             ;
                                51        B0   AD  9E 01F7E          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01F82          MOVAB   P.AJQ, R0                                        ;
                                    00000000G     00  16 01F89          JSB     DBG$CVT_DIVH2_R1                                 ;
                                51        C0   AD  9E 01F8F          MOVAB   INTMED_DATA+16, R1                               ;
                                50  00000000'    EF  9E 01F93          MOVAB   P.AJR, R0                                        ;
                                    00000000G     00  16 01F9A          JSB     DBG$CVT_DIVH2_R1                                 ;
                                        30        AE  D6 01FA0          INCL    SCALE                                            ;
                                                  D7  11 01FA3          BRB     415$                                             ;
                                                  58  D5 01FA5 416$:     TSTL    BIN_SCALE                                        ; 3327
                                                  26  15 01FA7          BLEQ    417$                                             ;
                                51        B0   AD  9E 01FA9          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01FAD          MOVAB   P.AJS, R0                                        ;
                                    00000000G     00  16 01FB4          JSB     DBG$CVT_MULH2_R1                                 ;
                                51        C0   AD  9E 01FBA          MOVAB   INTMED_DATA+16, R1                               ;
                                50  00000000'    EF  9E 01FBE          MOVAB   P.AJT, R0                                        ;
                                    00000000G     00  16 01FC5          JSB     DBG$CVT_MULH2_R1                                 ;
                                                  58  D7 01FCB          DECL    BIN_SCALE                                        ;
                                                  D6  11 01FCD          BRB     416$                                             ;
                                                  58  D5 01FCF 417$:     TSTL    BIN_SCALE                                        ;
                                                  26  18 01FD1          BGEQ    418$                                             ;
                                51        B0   AD  9E 01FD3          MOVAB   INTMED_DATA, R1                                  ;
                                50  00000000'    EF  9E 01FD7          MOVAB   P.AJU, R0                                        ;
                                    00000000G     00  16 01FDE          JSB     DBG$CVT_DIVH2_R1                                 ;
```

```
                        51      C0   AD  9E 01FE4          MOVAB   INTMED_DATA+16, R1                                    ;
                        50 00000000'EF  9E 01FE8          MOVAB   P.AJV, R0                                            ;
                           00000000G00  16 01FEF          JSB     DBG$CVT_DIVH2_R1                                     ;
                                   58  D6 01FF5          INCL    BIN_SCALE                                            ;
                                   D6  11 01FF7          BRB     417$                                                 ;
                                   30  AE D5 01FF9 418$:   TSTL    SCALE                                                ;
                                   27  15 01FFC          BLEQ    419$                                                 ;
                        51      B0   AD  9E 01FFE          MOVAB   INTMED_DATA, R1                                      ;
                        50 00000000'EF  9E 02002          MOVAB   P.AJW, R0                                            ;
                           00000000G00  16 02009          JSB     DBG$CVT_MULH2_R1                                     ;
                        51      C0   AD  9E 0200F          MOVAB   INTMED_DATA+16, R1                                   ;
                        50 00000000'EF  9E 02013          MOVAB   P.AJX, R0                                            ;
                           00000000G00  16 0201A          JSB     DBG$CVT_MULH2_R1                                     ;
                                   30  AE D7 02020          DECL    SCALE                                                ;
                                   D4  11 02023          BRB     418$                                                 ;
                                   27  18 02025 419$:   BGEQ    420$                                                 ;
                        51      B0   AD  9E 02027          MOVAB   INTMED_DATA, R1                                      ;
                        50 00000000'EF  9E 0202B          MOVAB   P.AJY, R0                                            ;
                           00000000G00  16 02032          JSB     DBG$CVT_DIVH2_R1                                     ;
                        51      C0   AD  9E 0203B          MOVAB   INTMED_DATA+16, R1                                   ;
                        50 00000000'EF  9E 0203C          MOVAB   P.AJZ, R0                                            ;
                           00000000G00  16 02043          JSB     DBG$CVT_DIVH2_R1                                     ;
                                   30  AE D6 02049          INCL    SCALE                                                ;
                                   D7  11 0204C          BRB     419$                                                 ;
                01              0A      6B  8F 0204E 420$:   CASEB   (R11), #10, #1                                       ; 3329
                              0057    0047    02052 421$:   .WORD   425$-421$,-                                          ;
                                                            427$-421$                                                    ;
                01              0C      6B  8F 02056          CASEB   (R11), #12, #1                                       ; 3339
                              0026    000D    0205A 422$:   .WORD   423$-422$,-                                          ;
                                                            424$-422$                                                    ;
                           00000000'EF  9F 0205E          PUSHAB  P.AKA                                                ; 3355
                                 0F62  31 02064          BRW     647$                                                 ;
                        50      B0   AD  9E 02067 423$:   MOVAB   INTMED_DATA, R0                                      ; 3344
                        51      34   AE  D0 0206B          MOVL    OUTPUT, R1                                           ;
                           00000000G00  16 0206F          JSB     DBG$CVT_CVTHF_R1                                     ;
51              34      AE              04  C1 02075          ADDL3   #4, OUTPUT, RT                                       ; 3345
                        50      C0   AD  9E 0207A          MOVAB   INTMED_DATA+16, R0                                   ;
                                   21  11 0207E          BRB     426$                                                 ;
                        50      B0   AD  9E 02080 424$:   MOVAB   INTMED_DATA, R0                                      ; 3350
                        51      34   AE  D0 02084          MOVL    OUTPUT, R1                                           ;
                           00000000G00  16 02088          JSB     DBG$CVT_CVTHD_R1                                     ;
51              34      AE              08  C1 0208E          ADDL3   #8, OUTPUT, RT                                       ; 3351
                        50      C0   AD  9E 02093          MOVAB   INTMED_DATA+16, R0                                   ;
                                   18  11 02097          BRB     428$                                                 ;
                        50      B0   AD  9E 02099 425$:   MOVAB   INTMED_DATA, R0                                      ; 3333
                        51      34   AE  D0 0209D          MOVL    OUTPUT, R1                                           ;
                           00000000G00  16 020A1 426$:   JSB     DBG$CVT_CVTHF_R1                                     ;
                                   0E  11 020A7          BRB     429$                                                 ;
                        50      B0   AD  9E 020A9 427$:   MOVAB   INTMED_DATA, R0                                      ; 3336
                        51      34   AE  D0 020AD          MOVL    OUTPUT, R1                                           ;
                           00000000G00  16 020B1 428$:   JSB     DBG$CVT_CVTHD_R1                                     ;
                                 0F1E  31 020B7 429$:   BRW     649$                                                 ; 2187
                        02              6B  91 020BA 430$:   CMPB    (R11), #2                                            ; 3364
                                   12  13 020BD          BEQL    433$                                                 ;
                        0E              6B  91 020BF          CMPB    (R11), #14                                           ;
                                   0D  13 020C2          BEQL    433$                                                 ;
                        25              6B  91 020C4          CMPB    (R11), #37                                           ;
```

```
                                        03  1E 020C7          BGEQU   432$                                                    ;
                                    0139  31 020C9 431$:      BRW     448$                                                    ;
                                27      6B 91 020CC 432$:     CMPB    (R11), #39                                              ;
                                        F8 1A 020CF           BGTRU   431$                                                    ;
                             58 AE      32 B0 020D1 433$:     MOVW    #50, CLASS_S_DESC                                       ; 3370
                             5C AE  60  AE 9E 020D5           MOVAB   TEMP_BUF2, CLASS_S_DESC+4                               ; 3371
                    01          1B  04  BE 8F 020DA           CASEB   @4(SP), #27, #1                                         ; 3372
                          004E          0004 020DF 434$:      .WORD   435$-434$,-
                                                              438$-434$                                                       ;
                                57        0F D0 020E3 435$:     MOVL    #15, DIGITS_IN_FRACT                                    ; 3376
                                52        03 D0 020E6           MOVL    #3, DIGITS_IN_EXP                                       ; 3377
                                50        07 D0 020E9           MOVL    #7, NOT_DIGITS                                          ; 3378
                50      F5  AD          10 00 ED 020EC          CMPZV   #0, #16, DST_INFO+5, NOT_DIGITS                         ; 3379
                                        15 15 020F2           BLEQ    437$                                                    ;
                                51      F5  AD 3C 020F4       MOVZWL  DST_INFO+5, R1                                          ; 3381
                                51        50 C2 020F8           SUBL2   NOT_DIGITS, R1                                          ;
                                50        57 D0 020FB           MOVL    DIGITS_IN_FRACT, R0                                     ;
                                51        50 D1 020FE           CMPL    R0, R1                                                  ;
                                        03 15 02101           BLEQ    436$                                                    ;
                                50        51 D0 02103           MOVL    R1, R0                                                  ;
                                57        50 D0 02106 436$:     MOVL    R0, DIGITS_IN_FRACT                                     ;
                                          52 DD 02109 437$:     PUSHL   DIGITS_IN_EXP                                           ; 3382
                                          7E D4 0210B           CLRL    -(SP)                                                   ;
                                      38  AE DD 0210D           PUSHL   SCALE                                                   ;
                                          57 DD 02110           PUSHL   DIGITS_IN_FRACT                                         ;
                                      68  AE 9F 02112           PUSHAB  CLASS_S_DESC                                            ;
                                      B0  AD 9F 02115           PUSHAB  INTMED_DATA                                             ;
                    00000000G     00        06 FB 02118           CALLS   #6, FOR$CVT_G_TE                                        ;
                                6E          50 D0 0211F           MOVL    R0, STATUS                                              ;
                                5F          6E E8 02122           BLBS    STATUS, 442$                                            ; 3383
                                  00000000' EF 9F 02125           PUSHAB  P.AKB                                                   ;
                                          48 11 0212B           BRB     441$                                                    ;
                                57        21 D0 0212D 438$:     MOVL    #33, DIGITS_IN_FRACT                                    ; 3388
                                52        04 D0 02130           MOVL    #4, DIGITS_IN_EXP                                       ; 3389
                                50        08 D0 02133           MOVL    #8, NOT_DIGITS                                          ; 3390
                50      F5  AD          10 00 ED 02136          CMPZV   #0, #16, DST_INFO+5, NOT_DIGITS                         ; 3391
                                        15 15 0213C           BLEQ    440$                                                    ;
                                51      F5  AD 3C 0213E       MOVZWL  DST_INFO+5, R1                                          ; 3393
                                51        50 C2 02142           SUBL2   NOT_DIGITS, R1                                          ;
                                50        57 D0 02145           MOVL    DIGITS_IN_FRACT, R0                                     ;
                                51        50 D1 02148           CMPL    R0, R1                                                  ;
                                        03 15 0214B           BLEQ    439$                                                    ;
                                50        51 D0 0214D           MOVL    R1, R0                                                  ;
                                57        50 D0 02150 439$:     MOVL    R0, DIGITS_IN_FRACT                                     ;
                                          52 DD 02153 440$:     PUSHL   DIGITS_IN_EXP                                           ; 3394
                                          7E D4 02155           CLRL    -(SP)                                                   ;
                                      38  AE DD 02157           PUSHL   SCALE                                                   ;
                                          57 DD 0215A           PUSHL   DIGITS_IN_FRACT                                         ;
                                      68  AE 9F 0215C           PUSHAB  CLASS_S_DESC                                            ;
                                      B0  AD 9F 0215F           PUSHAB  INTMED_DATA                                             ;
                    00000000G     00        06 FB 02162           CALLS   #6, FOR$CVT_H_TE                                        ;
                                6E          50 D0 02169           MOVL    R0, STATUS                                              ;
                                15          6E E8 0216C           BLBS    STATUS, 442$                                            ; 3395
                                  00000000' EF 9F 0216F           PUSHAB  P.AKC                                                   ;
                                          01 DD 02175 441$:     PUSHL   #1                                                      ;
                                  00028362  8F DD 02177           PUSHL   #164706                                                 ;
                    00000000G     00        03 FB 0217D           CALLS   #3, LIB$SIGNAL                                          ;
```

```
                60   AE          32              20  3B 02184 442$:   SKPC    #32, #50, TFMP_BUF2                     ; 3399
                                                 02  12 02189         BNEQ    443$
                                                 51  D4 0218B         CLRL    R1
                                     50      60  AE  9E 0218D 443$:   MOVAB   TEMP_BUF2, R0
                     5A              51          50  C3 02191         SUBL3   R0, R1, BUF_OFFSET
                     55              32          5A  C3 02195         SUBL3   BUF_OFFSET, #50, FINAL_LEN               ; 3400
                                 14  AE          55  D0 02199         MOVL    FINAL_LEN, OUTPUT_STR_LEN               ; 3401
                                     26          6B  91 0219D         CMPB    (R11), #38                              ; 3405
                                                 2E  12 021A0         BNEQ    445$
                                 58  AE          55  B0 021A2         MOVW    FINAL_LEN, CLASS_S_DESC                 ; 3409
                5C   AE          34  AE          01  C1 021A6         ADDL3   #1, OUTPUT, CLASS_S_DESC+4              ; 3410
                                     52      58  AE  9E 021AC         MOVAB   CLASS_S_DESC, R2                        ; 3411
                                     51      60 AE4A 9E 021B0         MOVAB   TEMP_BUF2[BUF_OFFSET], R1
                                     50          55  D0 021B5         MOVL    FINAL_LEN, R0
                                 00000000G       00  16 021B8         JSB     LIB$SCOPY_R_DX6
                                     6E          50  D0 021BE         MOVL    R0, STATUS
                         00000000G   8F          6E  D1 021C1         CMPL    STATUS, #LIB$_STRTRU                    ; 3412
                                                 03  13 021C8         BEQL    444$
                                               F4DA  31 021CA         BRW     319$
                                               F4C2  31 021CD 444$:   BRW     318$
                                     27          6B  91 021D0 445$:   CMPB    (R11), #39                              ; 3417
                                                 03  13 021D3         BEQL    446$
                                               F522  31 021D5         BRW     325$
                                 58  AE          55  B0 021D8 446$:   MOVW    FINAL_LEN, CLASS_S_DESC                 ; 3421
                                 5C  AE      34  AE  D0 021DC         MOVL    OUTPUT, CLASS_S_DESC+4                  ; 3422
                                     52      58  AE  9E 021E1         MOVAB   CLASS_S_DESC, R2                        ; 3423
                                     51      60 AE4A 9E 021E5         MOVAB   TEMP_BUF2[BUF_OFFSET], R1
                                     50          55  D0 021EA         MOVL    FINAL_LEN, R0
                                 00000000G       00  16 021ED         JSB     LIB$SCOPY_R_DX6
                                     6E          50  D0 021F3         MOVL    R0, STATUS
                         00000000G   8F          6E  D1 021F6         CMPL    STATUS, #LIB$_STRTRU                    ; 3424
                                                 03  13 021FD         BEQL    447$
                                               F4EF  31 021FF         BRW     323$
                                               F4D7  31 02202 447$:   BRW     322$
                                 00000000'       EF  9F 02205 448$:   PUSHAB  P.AKD                                  ; 3439
                                               0DBB  31 02208         BRW     647$
                                     50      FD  AD  3C 0220E 449$:   MOVZWL  SRC_INFO+5, R0                         ; 3447
                                 2C  AE          50  D0 02212         MOVL    R0, NO_DIGITS
                     58      0A  A9              01  03 EF 02216     EXTZV   #3, #1, 10(R9), R8                      ; 3445
                                                 6E  58 E9 0221C     BLBC    R8, 454$
                                     09      03  AA  91 0221F         CMPB    3(R10), #9                              ; 3447
                                                 0A  12 02223         BNEQ    450$
                                 30  AE      08  AA  98 02225         CVTBL   8(R10), SCALE
                                 30  AE      30  AE  CE 0222A         MNEGL   SCALE, SCALE
            FF7C CD      2C  AE      B0  AD      2C  AE  08 0222F 450$:   CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -
                                                                             TEMP_BUF1
                     58  AE          2C  AE          01  A1 02239         ADDW3   #1, NO_DIGITS, CLASS_S_DESC
                                     5C  AE  FF7C    CD  9E 0223F         MOVAB   TEMP_BUF1, CLASS_S_DESC+4
                                     7E      44      8F  9A 02245         MOVZBL  #68, -(SP)
                                             34      AE  DD 02249         PUSHL   SCALE
                                                     7E  D4 0224C         CLRL    -(SP)
                                             B0      AD  9F 0224E         PUSHAB  INTMED_DATA
                                             68      AE  9F 02251         PUSHAB  CLASS_S_DESC
                                 00000000G   00          05  FB 02254     CALLS   #5, OTS$CVT_T_H
                                     6E          50  D0 0225B         MOVL    R0, STATUS
                                     29          6E  E8 0225E         BLBS    STATUS, 453$
                                     50 00000000G    00  D0 02261         MOVL    DBG$GL_OPCODE_NAME, R0
```

```
                                                30  AE  D5 02268          TSTL    SCALE                                                   ;
                                                    0C  18 0226B          BGEQ    451$                                                    ;
                                                    50  DD 0226D          PUSHL   R0                                                      ;
                                                    01  DD 0226F          PUSHL   #1                                                      ;
                                          0002869B  8F  DD 02271          PUSHL   #165531                                                 ;
                                                    0A  11 02277          BRB     452$                                                    ;
                                                    50  DD 02279 451$:     PUSHL   R0                                                      ;
                                                    01  DD 0227B          PUSHL   #1                                                      ;
                                          00028A02  8F  DD 0227D          PUSHL   #166402                                                 ;
                                00000000G   00        03  FB 02283 452$:     CALLS   #3, LIB$SIGNAL                                      ;
                                                00D1  31  0228A 453$:     BRW     462$                                                    ;
08  BE          01      B0  AD      2C        AE  37 0228D 454$:     CMPP4   NO_DIGITS, INTMED_DATA, #1, @PACK_ZERO          ;3471
    54          54          02              02  EF 02297          EXTZV   #2, #2, R4, R4                                                  ;
                                            54  D7 0229C          DECL    R4                                                              ;
                                            04  15 0229E          BLEQ    455$                                                            ;
                            FF  AD          01  88 022A0          BISB2   #1, SRC_INFO+7                                                  ;
                                        30  AE  D5 022A4 455$:     TSTL    SCALE                                                           ;
                                            3C  13 022A7          BEQL    458$                                                            ;
            FF7C    CD      B0  AD      2C  AE  34 022A9          MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1                               ;
                                        0E  0C  AE  E9 022B1          BLBC    CVT_ROUND_FLAG, 456$                                            ;
                                        55  B0  AD  9E 022B5          MOVAB   INTMED_DATA, R5                                                 ;
                                        54  2C  AE  9E 022B9          MOVAB   NO_DIGITS, R4                                                   ;
                                    18  AE      05  D0 022BD          MOVL    #5, 24(SP)                                                      ;
                                            0B  11 022C1          BRB     457$                                                            ;
                                        55  B0  AD  9E 022C3 456$:     MOVAB   INTMED_DATA, R5                                                 ;
                                        54  2C  AE  9E 022C7          MOVAB   NO_DIGITS, R4                                                   ;
                                            18  AE  D4 022CB          CLRL    24(SP)                                                          ;
                                        53  18  AE  9E 022CE 457$:     MOVAB   24(SP), R3                                                      ;
                                        52  FF7C    CD  9E 022D2          MOVAB   TEMP_BUF1, R2                                               ;
                                        51  2C  AE  9E 022D7          MOVAB   NO_DIGITS, R1                                                   ;
                                        50  30  AE  9E 022DB          MOVAB   SCALE, R0                                                       ;
                                  00000000G 00  16  022DF          JSB     DBG$CVT_ASHP_R1                                                 ;
                                        50      6A  3C 022E5 458$:     MOVZWL  (R10), R0                                                       ;
                                        51  08  AA  98 022E8          CVTBL   8(R10), R1                                                      ;
                                        50      51  C0 022EC          ADDL2   R1, R0                                                          ;
                                        52      69  3C 022EF          MOVZWL  (R9), R2                                                        ;
                                        51  08  A9  98 022F2          CVTBL   8(R9), R1                                                       ;
                                        52      51  C0 022F6          ADDL2   R1, R2                                                          ;
                                        52      50  D1 022F9          CMPL    R0, R2                                                          ;
                                            58  15  022FC          BLEQ    461$                                                            ;
                                        15  04  BE  91 022FE          CMPB    @4(SP), #21                                                     ;
                                            52  12  02302          BNEQ    461$                                                            ;
                                        15      6B  91 02304          CMPB    (R11), #21                                                      ;
                                            4D  12  02307          BNEQ    461$                                                            ;
                                        50  BF  AD  9E 02309          MOVAB   INTMED_DATA+15, HIGH_NIBBLE_PTR                                 ;
                                        52      69  3C 0230D          MOVZWL  (R9), R2                                                        ;
                                        52      02  C6 02310          DIVL2   #2, R2                                                          ;
                                        52      50  C2 02313          SUBL2   HIGH_NIBBLE_PTR, R2                                             ;
                                        52      52  CE 02316          MNEGL   R2, LOW_NIBBLE_PTR                                              ;
                                        50      69  3C 02319          MOVZWL  (R9), R0                                                        ;
7E          00      50              01  7A 0231C          EMUL    #1, R0, #0, -(SP)                                                       ;
50          50      8E              02  7B 02321          EDIV    #2, (SP)+, R0, R0                                                       ;
                                            50  D5 02326          TSTL    R0                                                              ;
                                            08  12 02328          BNEQ    459$                                                            ;
50          62      04              00  EF 0232A          EXTZV   #0, #4, (LOW_NIBBLE_PTR), R0                                            ;
                    62              50  90 0232F          MOVB    R0, (LOW_NIBBLE_PTR)                                                    ;
```

```
                                           52  D7 02332 459$:  DECL    LOW_NIBBLE_PTR                              ;
                          50         B0    AD  9E 02334        MOVAB   INTMED DATA, R0                             ;
                          50               52  D1 02338        CMPL    LOW_NIBBLE_PTR, R0                          ;
                                           04  19 0233B        BLSS    460$                                        ;
                                           62  94 0233D        CLRB    (LOW_NIBBLE_PTR)                            ;
                                           F1  11 0233F        BRB     459$                                        ;
                              00000000G    00  DD 02341 460$:  PUSHL   DBG$GL_OPCODE_NAME                          ;
                                           01  DD 02347        PUSHL   #1                                          ;
                              0002809B     8F  DD 02349        PUSHL   #163995                                     ;
                    00000000G  00          03  FB 0234F        CALLS   #3, LIB$SIGNAL                              ;
          FF7C    CD      B0   AD    2C    AE  36 02356 461$:  CVTPL   NO_DIGITS, INTMED_DATA, TEMP_BUF1           ; 3473
                  29           01          6B  8F 0235E 462$:  CASEB   (RT1), #1, #41                              ; 3476
   0054           00BA         005D        023D   02362 463$:  .WORD   499$-463$,-                                 ;
   01CF           0173         0117        0054   0236A                465$-463$,-                                 ;
   0054           0054         0054        0054   02372                471$-463$,-                                 ;
   0054           0054         0054        0054   0237A                464$-463$,-                                 ;
   0054           0054         0054        0054   02382                464$-463$,-                                 ;
   0054           0054         0054        0054   0238A                478$-463$,-                                 ;
   0054           0054         0054        0054   02392                485$-463$,-                                 ;
   0054           0054         0054        0054   0239A                492$-463$,-                                 ;
   0054           0054         023D        0054   023A2                464$-463$,-                                 ;
   023D           0054         0054        0054   023AA                464$-463$,-                                 ;
                               023D        023D   023B2                464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       499$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       464$-463$,-                                 ;
                                                                       499$-463$,-                                 ;
                                                                       499$-463$,-                                 ;
                                                                       499$-463$                                   ;
                              00000000'    EF  9F 023B6 464$:  PUSHAB  P.AKE                                       ; 3713
                                   0C0A        31 023BC        BRW     647$                                        ;
                          03               58  E8 023BF 465$:  BLBS    R8, 466$                                    ;
```

```
                                                F88C   31  023C2                BRW     383$
                                          B0     AD   B5  023C5  466$:        TSTW    INTMED_DATA                                     ;3498
                                                 03   12  023C8                BNEQ    467$
                                               053C   31  023CA                BRW     542$
                                            FF7C   CD   D4  023CD  467$:        CLRL    TEMP_BUF1                                       ;3503
          56        B1    AD                      01   07  EF  023D1          EXTZV   #7, #1, INTMED_DATA+1, SIGN                      ;3504
                                      B1    AD     80   8F  8A  023D7          BICB2   #128, INTMED_DATA+1                             ;3505
                                            54     B0   AD  3C  023DC          MOVZWL  INTMED_DATA, FLOAT_SCALE                        ;3506
                                            54   C000   C4  9E  023E0          MOVAB   -16384(R4), FLOAT_SCALE
                                            52     08   A9  98  023E5          CVTBL   8(R9), R2                                       ;3507
                                            52         07  C0  023E9          ADDL2   #7, R2
                                            52         54  D1  023EC          CMPL    FLOAT_SCALE, R2
                                                       5B  14  023EF          BGTR    474$
                                      FF7C  CD       40   8F  88  023F1          BISB2   #64, TEMP_BUF1                                  ;3512
          50        B3    AD                      06   02  EF  023F7          EXTZV   #2, #6, INTMED_DATA+3, R0                       ;3513
    FF7C  CD              06                      00   50  F0  023FD          INSV    R0, #0, #6, TEMP_BUF1
                          54                      52   54  C3  02404          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                    ;3514
                                                       09  15  02408  468$:     BLEQ    469$                                            ;3515
                                      FF7C  CD             02  C6  0240A          DIVL2   #2, TEMP_BUF1                                   ;3517
                                                           54  D7  0240F          DECL    FLOAT_SCALE                                     ;3518
                                                           F5  11  02411          BRB     468$                                            ;3515
                                                    03     56  E8  02413  469$:     BLBS    SIGN, 470$                                      ;3520
                                                         0545  31  02416          BRW     549$
                                                         053B  31  02419  470$:     BRW     548$
                                                    03     58  E8  0241C  471$:     BLBS    R8, 472$                                        ;3532
                                                         F852  31  0241F          BRW     385$
                                                   B0    AD   B5  02422  472$:     TSTW    INTMED_DATA                                     ;3545
                                                         03   12  02425          BNEQ    473$
                                                       0554   31  02427          BRW     553$
                                                    FF7C   CD   D4  0242A  473$:     CLRL    TEMP_BUF1                                       ;3550
          56        B1    AD                      01   07  EF  0242E          EXTZV   #7, #1, INTMED_DATA+1, SIGN                      ;3551
                                      B1    AD     80   8F  8A  02434          BICB2   #128, INTMED_DATA+1                             ;3552
                                            54     B0   AD  3C  02439          MOVZWL  INTMED_DATA, FLOAT_SCALE                        ;3553
                                            54   C000   C4  9E  0243D          MOVAB   -16384(R4), FLOAT_SCALE
                                            52     08   A9  98  02442          CVTBL   8(R9), R2                                       ;3554
                                            52         0F  C0  02446          ADDL2   #15, R2
                                            52         54  D1  02449          CMPL    FLOAT_SCALE, R2
                                                       60  14  0244C  474$:     BGTR    482$
                                      FF7D  CD       40   8F  88  0244E          BISB2   #64, TEMP_BUF1+1                                ;3559
          50        B2    AD                      0E   02  EF  02454          EXTZV   #2, #14, INTMED_DATA+2, R0                      ;3560
    FF7C  CD              0E                      00   50  F0  0245A          INSV    R0, #0, #14, TEMP_BUF1
                          54                      52   54  C3  02461          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                    ;3561
                                                       09  15  02465  475$:     BLEQ    476$                                            ;3562
                                      FF7C  CD             02  C6  02467          DIVL2   #2, TEMP_BUF1                                   ;3564
                                                           54  D7  0246C          DECL    FLOAT_SCALE                                     ;3565
                                                           F5  11  0246E          BRB     475$                                            ;3562
                                                    03     56  E8  02470  476$:     BLBS    SIGN, 477$                                      ;3567
                                                         055D  31  02473          BRW     560$
                                                         0553  31  02476  477$:     BRW     559$
                                                    08     58  E8  02479  478$:     BLBS    R8, 480$                                        ;3579
                                                    50  FF7C   CD  9E  0247C  479$:     MOVAB   TEMP_BUF1, R0                           ;3581
                                                         0474  31  02481          BRW     540$
                                                   B0    AD   B5  02484  480$:     TSTW    INTMED_DATA                                     ;3587
                                                         03   12  02487          BNEQ    481$
                                                       047D   31  02489          BRW     542$
                                                    FF7C   CD   D4  0248C  481$:     CLRL    TEMP_BUF1                                       ;3592
          56        B1    AD                      01   07  EF  02490          EXTZV   #7, #1, INTMED_DATA+1, SIGN                      ;3593
```

```
                        B1   AD      80   8F   BA 02496          BICB2   #128, INTMED_DATA+1                          ; 3594
                             54      B0   AD   3C 0249B          MOVZWL  INTMED_DATA, FLOAT_SCALE                     ; 3595
                             54    C000   C4   9E 0249F          MOVAB   -16384(R4), FLOAT_SCALE
                             52      08   A9   98 024A4          CVTBL   8(R9), R2                                    ; 3596
                             52           07   C0 024A8          ADDL2   #7, R2
                             52           54   D1 024AB          CMPL    FLOAT_SCALE, R2
                                          5A   14 024AE 482$:     BGTR    489$
                        FF7C CD          40   8F   88 024B0          BISB2   #64, TEMP_BUF1                           ; 3601
        50      B3   AD      06           02   EF 024B6          EXTZV   #2, #6, INTMED_DATA+3, R0                    ; 3602
FF7C    CD           06      00           50   F0 024BC          INSV    R0, #0, #6, TEMP_BUF1
                     54      52           54   C3 024C3          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                 ; 3603
                                          03   14 024C7 483$:     BGTR    484$                                         ; 3604
                                        FF47   31 024C9          BRW     469$
                        FF7C CD           02   C6 024CC 484$:     DIVL2   #2, TEMP_BUF1                            ; 3606
                                          54   D7 024D1          DECL    FLOAT_SCALE                                  ; 3607
                                          F2   11 024D3          BRB     483$                                         ; 3604
                                     08   58   E8 024D5 485$:     BLBS    R8, 487$                                     ; 3621
                                     50 FF7C   CD   9E 024D8 486$:     MOVAB   TEMP_BUF1, R0                        ; 3623
                                        048D   31 024DD          BRW     551$
                                   B0   AD   B5 024E0 487$:     TSTW    INTMED_DATA                                  ; 3629
                                        03   12 024E3          BNEQ    488$
                                        0496   31 024E5          BRW     553$
                                    FF7C   CD   D4 024E8 488$:     CLRL    TEMP_BUF1                                ; 3634
        56      B1   AD      01           07   EF 024EC          EXTZV   #7, #1, INTMED_DATA+1, SIGN                  ; 3635
                        B1   AD      80   8F   8A 024F2          BICB2   #128, INTMED_DATA+1                          ; 3636
                             54      B0   AD   3C 024F7          MOVZWL  INTMED_DATA, FLOAT_SCALE                     ; 3637
                             54    C000   C4   9E 024FB          MOVAB   -16384(R4), FLOAT_SCALE
                             52      08   A9   98 02500          CVTBL   8(R9), R2                                    ; 3638
                             52           0F   C0 02504          ADDL2   #15, R2
                             52           54   D1 02507          CMPL    FLOAT_SCALE, R2
                                          52   14 0250A 489$:     BGTR    494$
                        FF7D CD          40   8F   88 0250C          BISB2   #64, TEMP_BUF1+1                         ; 3643
        50      B2   AD      0E           02   EF 02512          EXTZV   #2, #14, INTMED_DATA+2, R0                   ; 3644
FF7C    CD           0E      00           50   F0 02518          INSV    R0, #0, #14, TEMP_BUF1
                     54      52           54   C3 0251F          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                 ; 3645
                                          03   14 02523 490$:     BGTR    491$                                         ; 3646
                                        FF48   31 02525          BRW     476$
                        FF7C CD           02   C6 02528 491$:     DIVL2   #2, TEMP_BUF1                            ; 3648
                                          54   D7 0252D          DECL    FLOAT_SCALE                                  ; 3649
                                          F2   11 0252F          BRB     490$                                         ; 3646
                                     63   58   E9 02531 492$:     BLBC    R8, 498$                                     ; 3663
                                   B0   AD   B5 02534          TSTW    INTMED_DATA                                  ; 3671
                                        03   12 02537          BNEQ    493$
                                        04A8   31 02539          BRW     562$
                                    FF7C   CD   D4 0253C 493$:     CLRL    TEMP_BUF1                                ; 3676
        56      B1   AD      01           07   EF 02540          EXTZV   #7, #1, INTMED_DATA+1, SIGN                  ; 3677
                        B1   AD      80   8F   8A 02546          BICB2   #128, INTMED_DATA+1                          ; 3678
                             54      B0   AD   3C 0254B          MOVZWL  INTMED_DATA, FLOAT_SCALE                     ; 3679
                             54    C000   C4   9E 0254F          MOVAB   -16384(R4), FLOAT_SCALE
                             52      08   A9   98 02554          CVTBL   8(R9), R2                                    ; 3680
                             52           1F   C0 02558          ADDL2   #31, R2
                             52           54   D1 0255B          CMPL    FLOAT_SCALE, R2
                                          03   15 0255E 494$:     BLEQ    495$
                                        04AC   31 02560          BRW     565$
                        FF7F CD          40   8F   88 02563 495$:     BISB2   #64, TEMP_BUF1+3                         ; 3685
FF7D    CD           10      06           B2   AD   F0 02569          INSV    INTMED_DATA+2, #6, #16, TEMP_BUF1+1      ; 3686
        50      B6   AD      0E           02   EF 02571          EXTZV   #2, #14, INTMED_DATA+6, R0                   ; 3687
```

```
 FF7C   CD        0E         00          50 F0 02577        INSV    R0, #0, #14, TEMP_BUF1
                  54         52          54 C3 0257E        SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE          3688
                             09          15 02582  496$:    BLEQ    497$                                  3689
                      FF7C   CD          02 C6 02584        DIVL2   #2, TEMP_BUF1                         3691
                             54          D7 02589           DECL    FLOAT_SCALE                           3692
                             F5          11 0258B           BRB     496$                                  3689
                             07          56 E9 0258D  497$: BLBC    SIGN, 498$                            3694
                      FF7C   CD    FF7C  CD CE 02590        MNEGL   TEMP_BUF1, TEMP_BUF1
                        34   BE    FF7C  CD D0 02597  498$: MOVL    TEMP_BUF1, @OUTPUT                    3695
                                         19 11 0259D        BRB     502$                                  3476
                             52    F5    AD 3C 0259F  499$: MOVZWL  DST_INFO+5, R2                        3706
                             50          01 CE 025A3        MNEGL   #1, I                                 3708
                                         0C 11 025A6        BRB     501$
          51      B0  AD     01          50 EF 025A8  500$: EXTZV   I, #1, INTMED_DATA, R1
    34    BE          01     50          51 F0 025AE        INSV    R1, I, #1, @OUTPUT
                      F0     50          52 F2 025B4  501$: AOBLSS  R2, I, 500$                           3706
                                   0A1D  31 025B8     502$: BRW     649$                                  2187
                        2C   AE    FD    AD 3C 025BB  503$: MOVZWL  SRC_INFO+5, NO_DIGITS                 3718
    08    BE          01     B0    AD 2C AE 37 025C0        CMPP4   NO_DIGITS, INTMED_DATA, #1, @PACK_ZERO
                                         54 DC 025C8        MOVPSL  R4
          54          54     02          02 EF 025CA        EXTZV   #2, #2, R4, R4
                                         54 D7 025CF        DECL    R4
                                         04 15 025D1        BLEQ    504$
                        FF   AD          01 88 025D3        BISB2   #1, SRC_INFO+7
                                   30    AE D5 025D7  504$: TSTL    SCALE
                                         3C 13 025DA        BEQL    507$
          FF7C    CD  B0     AD    2C    AE 34 025DC        MOVP    NO_DIGITS, INTMED_DATA, TEMP_BUF1
                             0E    0C    AE E9 025E4        BLBC    CVT_ROUND_FLAG, 505$
                             55    B0    AD 9E 025E8        MOVAB   INTMED_DATA, R5
                             54    2C    AE 9E 025EC        MOVAB   NO_DIGITS, R4
                        18   AE          05 D0 025F0        MOVL    #5, 24(SP)
                                         08 11 025F4        BRB     506$
                             55    B0    AD 9E 025F6  505$: MOVAB   INTMED_DATA, R5
                             54    2C    AE 9E 025FA        MOVAB   NO_DIGITS, R4
                                   18    AE D4 025FE        CLRL    24(SP)
                             53    18    AE 9E 02601  506$: MOVAB   24(SP), R3
                             52  FF7C    CD 9E 02605        MOVAB   TEMP_BUF1, R2
                             51    2C    AE 9E 0260A        MOVAB   NO_DIGITS, R1
                             50    30    AE 9E 0260E        MOVAB   SCALE, R0
                           00000000G     00 16 02612        JSB     DBG$CVT_ASHP_R1
                             50          6A 3C 02618  507$: MOVZWL  (R10), R0
                             51    08    AA 98 0261B        CVTBL   8(R10), R1
                             50          51 C0 0261F        ADDL2   R1, R0
                             52          69 3C 02622        MOVZWL  (R9), R2
                             51    08    A9 98 02625        CVTBL   8(R9), R1
                             52          51 C0 02629        ADDL2   R1, R2
                             52          50 D1 0262C        CMPL    R0, R2
                                         58 15 0262F        BLEQ    510$
                             15    04    BE 91 02631        CMPB    @4(SP), #21
                                         52 12 02635        BNEQ    510$
                             15          6B 91 02637        CMPB    (R11), #21
                                         4D 12 0263A        BNEQ    510$
                             50    BF    AD 9E 0263C        MOVAB   INTMED_DATA+15, HIGH_NIBBLE_PTR
                             52          69 3C 02640        MOVZWL  (R9), R2
                             52          02 C6 02643        DIVL2   #2, R2
                             52          50 C2 02646        SUBL2   HIGH_NIBBLE_PTR, R2
                             52          52 CE 02649        MNEGL   R2, LOW_NIBBLE_PTR
```

```
                                                50              69  3C 0264C                MOVZWL  (R9), R0                                        ;
                                7E      00      50              01  7A 0264F                EMUL    #1, R0, #0, -(SP)                               ;
                                50      50      8E              02  7B 02654                EDIV    #2, (SP)+, R0, R0                               ;
                                                                50  D5 02659                TSTL    R0                                              ;
                                                                08  12 0265B                BNEQ    508$                                            ;
                                50      62              04      00  EF 0265D                EXTZV   #0, #4, (LOW_NIBBLE_PTR), R0                    ;
                                                        62      50  90 02662                MOVB    R0, (LOW_NIBBLE_PTR)                            ;
                                                                52  D7 02665 508$:          DECL    LOW_NIBBLE_PTR                                  ;
                                                50      B0      AD  9E 02667                MOVAB   INTMED_DATA, R0                                 ;
                                                50              52  D1 0266B                CMPL    LOW_NIBBLE_PTR, R0                              ;
                                                                04  19 0266E                BLSS    509$                                            ;
                                                                62  94 02670                CLRB    (LOW_NIBBLE_PTR)                                ;
                                                                F1  11 02672                BRB     508$                                            ;
                                                00000000G       00  DD 02674 509$:          PUSHL   DBG$GL_OPCODE_NAME                              ;
                                                                01  DD 0267A                PUSHL   #1                                              ;
                                                0002809B        8F  DD 0267C                PUSHL   #163995                                         ;
                                        00000000G       00      03  FB 02682                CALLS   #3, LIB$SIGNAL                                  ;
                                    16          04              6B  8F 02689 510$:          CASEB   (R11), #4, #22                                  ; 3721
                            002E        002E        006C        0037    02680 511$:         .WORD   513$-511$,-                                     ;
                            002E        002E        006C        002E    02695               513$-511$,-                                             ;
                            002E        002E        002E        002E    0269D               512$-511$,-                                             ;
                            002E        002E        002E        002E    026A5               512$-511$,-                                             ;
                            002E        002E        002E        002E    026AD               512$-511$,-                                             ;
                            006C        002E        002E        002E    026B5               515$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            512$-511$,-                                             ;
                                                                                            515$-511$                                               ;
                                                00000000'       EF  9F 026BB 512$:          PUSHAB  P.AKF                                           ; 3749
                                                        0905    31  C1 026C1                BRW     647$                                            ;
                1C  BE          0A      B0      AD      2C      AE  37 026C4 513$:          CMPP4   NO_DIGITS, INTMED_DATA, #10, @LRGST_P_LU        ; 3726
                    58          58              02              58  DC 026CC                MOVPSL  R8                                              ;
                                                15              02  EF 026CE                EXTZV   #2, #2, R8, R8                                  ;
                                                                58  F5 026D3                SOBGTR  R8, 514$                                        ;
                                                00000000G       00  DD 026D6                PUSHL   DBG$GL_OPCODE_NAME                              ; 3728
                                                                01  DD 026DC                PUSHL   #1                                              ;
                                                000286A3        8F  DD 026DE                PUSHL   #165539                                         ;
                                        00000000G       00      03  FB 026E4                CALLS   #3, LIB$SIGNAL                                  ;
                                                        00E0    8F  B9 026EB 514$:          BICPSW  #224                                            ; 3729
                            34  BE          B0      AD      2C      AE  36 026EF                CVTPL   NO_DIGITS, INTMED_DATA, @OUTPUT                 ; 3730
                                                        0448    31  F6 026F6                BRW     583$                                            ; 3731
                60  AE      2C  AE          B0      AD      2C      AE  08 026F9 515$:          CVTPS   NO_DIGITS, INTMED_DATA, NO_DIGITS, -             ; 3736
                                                                                                    TEMP_BUF2                                       ;
                            58  AE          2C      AE              01  A1 02702                ADDW3   #1, NO_DIGITS, CLASS_S_DESC                   ; 3737
```

```
                      5C   AE       60   AE  9E 02708          MOVAB   TEMP_BUF2, CLASS_S_DESC+4           ; 3738
                                  FF7C   CD  9F 0270D          PUSHAB  TEMP_BUF1                           ; 3739
                                    5C   AE  9F 02711          PUSHAB  CLASS_S_DESC                        ;
               00000000G   00            02  FB 02714          CALLS   #2, OTS$CVT_T_H                     ;
                           09            6B  91 0271B          CMPB    (R11), #9                           ; 3740
                                         03  13 0271E          BEQL    516$                                ;
                           05            6B  91 02720          CMPB    (R11), #5                           ; 3741
                                         03  12 02723 516$:    BNEQ    517$                                ;
                                       045C  31 02725          BRW     585$                                ;
                           51       34   AE  9E 02728 517$:    MOVAB   OUTPUT, R1                          ; 3745
                           50     FF7C   CD  9E 0272C          MOVAB   TEMP_BUF1, R0                       ;
                             00000000G   00  16 02731 518$:    JSB     DBG$CVT_CVTRHO_R1                   ;
                                       089E  31 02737          BRW     649$                                ; 3721
                      58   AE       FD   AD  B0 0273A 519$:    MOVW    SRC_INFO+5, CLASS_S_DESC            ; 3755
                      5C   AE       F9   AD  D0 0273F          MOVL    SRC_INFO+1, CLASS_S_DESC+4          ; 3756
                           7E       55   8F  9A 02744          MOVZBL  #85, -(SP)                          ; 3758
                           7E       34   AE  CE 02748          MNEGL   SCALE, -(SP)                        ; 3757
                                         7E  D4 0274C          CLRL    -(SP)                               ;
                                  FF7C   CD  9F 0274E          PUSHAB  TEMP_BUF1                           ;
                                    68   AE  9F 02752          PUSHAB  CLASS_S_DESC                        ;
               00000000G   00            05  FB 02755          CALLS   #5, OTS$CVT_T_H                     ;
                           6E            50  D0 0275C          MOVL    R0, STATUS                          ;
                           15            6E  E8 0275F          BLBS    STATUS, 520$                        ; 3759
                             00000000G   00  DD 02762          PUSHL   DBG$GL_OPCODE_NAME                  ;
                                         01  DD 02768          PUSHL   #1                                  ;
                              00028298   8F  DD 0276A          PUSHL   #164504                             ;
               00000000G   00            03  FB 02770          CALLS   #3, LIB$SIGNAL                      ;
    52        0A   A9      01            03  EF 02777 520$:    EXTZV   #3, #1, 10(R9), R2                  ; 3780
                           1C            52  E8 0277D          BLBS    R2, 522$                            ;
                           11       0C   AE  E9 02780          BLBC    CVT_ROUND_FLAG, 521$                ; 3782
                           51       60   AE  9E 02784          MOVAB   TEMP_BUF2, R1                       ; 3784
                           50     FF7C   CD  9E 02788          MOVAB   TEMP_BUF1, R0                       ;
                             00000000G   00  16 0278D          JSB     DBG$CVT_CVTRHL_R1                   ;
                                         07  11 02793          BRB     522$                                ;
                      60   AE     FF7C   CD6AFD 02795 521$:    CVTHL   TEMP_BUF1, TEMP_BUF2                ; 3786
                      29   01            6B  8F 0279C 522$:    CASEB   (R11), #1, #41                      ; 3788
    0054      00D7        005D        02B6      027A0 523$:    .WORD   571$-523$,-                         ;
    023B      01C6        0151        0054      027A8                  525$-523$,-                         ;
    0054      0054        0054        0054      027B0                  531$-523$,-                         ;
    0054      0054        0054        0054      027B8                  524$-523$,-                         ;
    0054      0054        0054        0054      027C0                  524$-523$,-                         ;
    0054      0054        0054        0054      027C8                  539$-523$,-                         ;
    0054      0054        0054        0054      027D0                  550$-523$,-                         ;
    0054      0054        0054        0054      027D8                  561$-523$,-                         ;
    0054      0054        02B6        0054      027E0                  524$-523$,-                         ;
    02B6      0054        0054        0054      027E8                  524$-523$,-                         ;
                          02B6        02B6      027F0                  524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
                                                                       524$-523$,-                         ;
```

```
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       571$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       524$-523$,-                       ;
                                                                       571$-523$,-                       ;
                                                                       571$-523$,-                       ;
                                                                       571$-523$                         ;
                          00000000'  EF  9F 027F4 524$:   PUSHAB  P.AKG                                  ; 4025
                               07CC  31 027FA            BRW     647$                                    ;
                   26          52  E8 027FD 525$:   BLBS    R2, 527$                                     ;
          000000FF 8F     60  AE  D1 02800            CMPL    TEMP_BUF2, #255                              ; 3800
                               15  1B 02808            BLEQU   526$                                      ;
                          00000000G  00  DD 0280A            PUSHL   DBG$GL_OPCODE_NAME                  ; 3802
                               01  DD 02810            PUSHL   #1                                        ;
                          000286A3  8F  DD 02812            PUSHL   #165539                              ;
          00000000G 00         03  FB 02818            CALLS   #3, LIB$SIGNAL                            ;
                 34 BE    60  AE  90 0281F 526$:   MOVB    TEMP_BUF2, @OUTPUT                            ; 3803
                               78  11 02824            BRB     533$                                      ; 3797
                          B0  AD  B5 02826 527$:   TSTW    INTMED_DATA                                   ; 3810
                               03  12 02829            BNEQ    528$                                      ;
                             00DB  31 028[illegible]            BRW     542$                             ;
                        FF7C  CD  D4 0282E 528$:   CLRL    TEMP_BUF1                                     ; 3815
    56     B1   AD       01    07  EF 02832            EXTZV   #7, #1, INTMED_DATA+1, SIGN               ; 3816
                   B1    AD   80  8F  8A 02838            BICB2   #128, INTMED_DATA+1                    ; 3817
                         54   B0  AD  3C 0283D            MOVZWL  INTMED_DATA, FLOAT_SCALE               ; 3818
                         54 C000  C4  9E 02841            MOVAB   -16384(R4), FLOAT_SCALE                ;
                         52   08  A9  98 02846            CVTBL   8(R9), R2                              ; 3819
                         52       07  C0 0284A            ADDL2   #7, R2                                 ;
                         52       54  D1 0284D            CMPL    FLOAT_SCALE, R2                        ;
                                  78  14 02850            BGTR    536$                                   ;
                 FF7C    CD   40  8F  88 02852            BISB2   #64, TEMP_BUF1                         ; 3824
    50     B3   AD       06       02  EF 02858            EXTZV   #2, #6, INTMED_DATA+3, R0              ; 3825
FF7C  CD        06       00       50  F0 0285E            INSV    R0, #0, #6, TEMP_BUF1                  ;
                54       52       54  C3 02865            SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE           ; 3826
                                  03  14 02869 529$:   BGTR    530$                                      ; 3827
                                FBA5  31 0286B            BRW     469$                                   ;
                 FF7C    CD       02  C6 0286E 530$:   DIVL2   #2, TEMP_BUF1                             ; 3829
                                  54  D7 02873            DECL    FLOAT_SCALE                            ; 3830
                                  F2  11 02875            BRB     529$                                   ; 3827
                         26       52  E8 02877 531$:   BLBS    R2, 534$                                  ; 3844
          0000FFFF       8F   60  AE  D1 0287A            CMPL    TEMP_BUF2, #65535                      ; 3847
                                  15  1B 02882            BLEQU   532$                                   ;
                          00000000G  00  DD 02884            PUSHL   DBG$GL_OPCODE_NAME                  ; 3849
```

```
                                                 01   DD 0288A          PUSHL   #1                                                    ;
                                      000286A3   8F   DD 0288C          PUSHL   #165539                                               ;
                          00000000G   00         03   FB 02892          CALLS   #3, LIB$SIGNAL                                        ;
                                 34   BE    60   AE   B0 02899 532$:    MOVW    TEMP_BUF2, @OUTPUT                                    ;3850
                                                 6C   11 0289E 533$:    BRB     543$                                                  ;3844
                                      B0    AD   B5 028A0 534$:    TSTW    INTMED_DATA                                           ;3857
                                            03   12 028A3               BNEQ    535$                                                  ;
                                          00D6   31 028A5               BRW     553$                                                  ;
                                      FF7C  CD   D4 028A8 535$:    CLRL    TEMP_BUF1                                             ;3862
                    56    B1    AD              07   EF 028AC          EXTZV   #7, #1, INTMED_DATA+1, SIGN                           ;3863
                                 01   AD    80   8F   8A 028B2          BICB2   #128, INTMED_DATA+1                                   ;3864
                                 54   B0    AD   3C 028B7               MOVZWL  INTMED_DATA, FLOAT_SCALE                              ;3865
                                 54   C000  C4   9E 028BB               MOVAB   -16384(R4), FLOAT_SCALE                               ;
                                 52   08    A9   98 028C0               CVTBL   8(R9), R2                                             ;3866
                                 52         0F   C0 028C4               ADDL2   #15, R2                                               ;
                                 52         54   D1 028C7               CMPL    FLOAT_SCALE, R2                                       ;
                                            64   14 028CA 536$:    BGTR    545$                                                  ;
                                      FF7D  CD   40   8F   88 028CC          BISB2   #64, TEMP_BUF1+1                                      ;3871
                    50    B2    AD              0E   02   EF 028D2          EXTZV   #2, #14, INTMED_DATA+2, R0                            ;3872
              FF7C  CD          0E              00   50   F0 028D8          INSV    R0, #0, #14, TEMP_BUF1                                ;
                          54                    52   54   C3 028DF          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                          ;3873
                                            03   14 028E3 537$:    BGTR    538$                                                  ;3874
                                          FB88   31 028E5               BRW     476$                                                  ;
                                      FF7C  CD         02   C6 028E8 538$:    DIVL2   #2, TEMP_BUF1                                         ;3876
                                            54   D7 028ED               DECL    FLOAT_SCALE                                           ;3877
                                            F2   11 028EF               BRB     537$                                                  ;3874
                                 10         52   E8 028F1 539$:    BLBS    R2, 541$                                              ;3891
                                 50    60   AE   9E 028F4               MOVAB   TEMP_BUF2, R0                                         ;3893
                                 51    34   AE   D0 028F8 540$:    MOVL    OUTPUT, R1                                            ;
                                      00000000G  00   16 028FC          JSB     DBG$CVT_CVTLB_R1                                      ;
                                            7D   11 02902               BRB     554$                                                  ;
                                      B0    AD   B5 02904 541$:    TSTW    INTMED_DATA                                           ;3899
                                            05   12 02907               BNEQ    544$                                                  ;
                                      34    BE   94 02909 542$:    CLRB    @OUTPUT                                               ;3901
                                            73   11 0290C 543$:    BRB     554$                                                  ;
                                      FF7C  CD   D4 0290E 544$:    CLRL    TEMP_BUF1                                             ;3904
                    56    B1    AD              07   EF 02912          EXTZV   #7, #1, INTMED_DATA+1, SIGN                           ;3905
                                 B1   AD    80   8F   8A 02918          BICB2   #128, INTMED_DATA+1                                   ;3906
                                 54   B0    AD   3C 0291D               MOVZWL  INTMED_DATA, FLOAT_SCALE                              ;3907
                                 54   C000  C4   9E 02921               MOVAB   -16384(R4), FLOAT_SCALE                               ;
                                 52   08    A9   98 02926               CVTBL   8(R9), R2                                             ;3908
                                 52         07   C0 0292A               ADDL2   #7, R2                                                ;
                                 52         54   D1 0292D               CMPL    FLOAT_SCALE, R2                                       ;
                                            73   14 02930 545$:    BGTR    556$                                                  ;
                                      FF7C  CD   40   8F   88 02932          BISB2   #64, TEMP_BUF1                                        ;3913
                    50    B3    AD              06   02   EF 02938          EXTZV   #2, #6, INTMED_DATA+3, R0                             ;3914
              FF7C  CD          06              00   50   F0 0293E          INSV    R0, #0, #6, TEMP_BUF1                                 ;
                          54                    52   54   C3 02945          SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE                          ;3915
                                            09   15 02949 546$:    BLEQ    547$                                                  ;3916
                                      FF7C  CD         02   C6 0294B          DIVL2   #2, TEMP_BUF1                                         ;3918
                                            54   D7 02950               DECL    FLOAT_SCALE                                           ;3919
                                            F5   11 02952               BRB     546$                                                  ;3916
                                 07         56   E9 02954 547$:    BLBC    SIGN, 549$                                            ;3921
                                      FF7C  CD   FF7C  CD   CE 02957 548$:    MNEGL   TEMP_BUF1, TEMP_BUF1                                  ;
                                      34    BE   FF7C  CD   90 0295E 549$:    MOVB    TEMP_BUF1, @OUTPUT                            ;3922
                                            1B   11 02964               BRB     554$                                                  ;3788
                                 10         52   E8 02966 550$:    BLBS    R2, 552$                                              ;3933
```

```
                          50          60  AE  9E 02969         MOVAB   TEMP_BUF2, R0                            ; 3935
                          51          34  AE  D0 0296D 551$:   MOVL    OUTPUT, R1                               ;
                             00000000G    00  16 02971         JSB     DBG$CVT_CVTLW_R1                         ;
                                          6E  11 02977         BRB     563$                                     ;
                                      B0  AD  B5 02979 552$:   TSTW    INTMED_DATA                              ; 3941
                                          05  12 0297C         BNEQ    555$                                     ;
                                      34  BE  B4 0297E 553$:   CLRW    @OUTPUT                                  ; 3943
                                          64  11 02981 554$:   BRB     563$                                     ;
                                    FF7C  CD  D4 02983 555$:   CLRL    TEMP_BUF1                                ; 3946
        56        B1   AD         01      07  EF 02987         EXTZV   #7, #1, INTMED_DATA+1, SIGN              ; 3947
                           B1     AD  80  8F  8A 0298D         BICB2   #128, INTMED_DATA+1                      ; 3948
                                  54  B0  AD  3C 02992         MOVZWL  INTMED_DATA, FLOAT_SCALE                 ; 3949
                                  54 C000 C4  9E 02996         MOVAB   -16384(R4), FLOAT_SCALE                  ;
                                  52  08  A9  98 0299B         CVTBL   8(R9), R2                                ; 3950
                                  52      0F  C0 0299F         ADDL2   #15, R2                                  ;
                                  52      54  D1 029A2         CMPL    FLOAT_SCALE, R2                          ;
                                          68  14 029A5 556$:   BGTR    565$                                     ;
                         FF7D     CD  40  8F  88 029A7         BISB2   #64, TEMP_BUF1+1                         ; 3955
        50        B2   AD         0E      02  EF 029AD         EXTZV   #2, #14, INTMED_DATA+2, R0               ; 3956
FF7C    CD             0E         00      50  F0 029B3         INSV    R0, #0, #14, TEMP_BUF1                   ;
                       54         52      54  C3 029BA         SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE             ; 3957
                                          09  15 029BE 557$:   BLEQ    558$                                     ; 3958
                         FF7C     CD      02  C6 029C0         DIVL2   #2, TEMP_BUF1                            ; 3960
                                          54  D7 029C5         DECL    FLOAT_SCALE                              ; 3961
                                          F5  11 029C7         BRB     557$                                     ; 3958
                                  07      56  E9 029C9 558$:   BLBC    SIGN, 560$                               ; 3963
                         FF7C     CD FF7C CD  CE 029CC 559$:   MNEGL   TEMP_BUF1, TEMP_BUF1                     ;
                           34     BE FF7C CD  B0 029D3 560$:   MOVW    TEMP_BUF1, @OUTPUT                       ; 3964
                                          79  11 029D9         BRB     570$                                     ; 3788
                                  71      52  E9 029DB 561$:   BLBC    R2, 569$                                 ; 3975
                                    FF7C  CD  B5 029DE         TSTW    TEMP_BUF1                                ; 3983
                                          05  12 029E2         BNEQ    564$                                     ;
                                      34  BE  D4 029E4 562$:   CLRL    @OUTPUT                                  ; 3985
                                          6B  11 029E7 563$:   BRB     570$                                     ;
                                      60  AE  D4 029E9 564$:   CLRL    TEMP_BUF2                                ; 3988
        56      FF7D   CD         01      07  EF 029EC         EXTZV   #7, #1, TEMP_BUF1+1, SIGN                ; 3989
                         FF7D     CD  80  8F  8A 029F3         BICB2   #128, TEMP_BUF1+1                        ; 3990
                                  54 FF7C CD  3C 029F9         MOVZWL  TEMP_BUF1, FLOAT_SCALE                   ; 3991
                                  54 C000 C4  9E 029FE         MOVAB   -16384(R4), FLOAT_SCALE                  ;
                                  52  08  A9  98 02A03         CVTBL   8(R9), R2                                ; 3992
                                  52      1F  C0 02A07         ADDL2   #31, R2                                  ;
                                  52      54  D1 02A0A         CMPL    FLOAT_SCALE, R2                          ;
                                          11  15 02A0D         BLEQ    566$                                     ;
                             00000000G    00  DD 02A0F 565$:   PUSHL   DBG$GL_OPCODE_NAME                       ; 3994
                                          01  DD 02A15         PUSHL   #1                                       ;
                             000286A3     8F  DD 02A17         PUSHL   #165539                                  ;
                                     05B1 31 02A1D             BRW     648$                                     ;
                           63     AE  40  8F  88 02A20 566$:   BISB2   #64, TEMP_BUF2+3                         ; 3997
  61    AE             10         06 FF7E CD  F0 02A25         INSV    TEMP_BUF1+2, #6, #16, TEMP_BUF2+1        ; 3998
        50        82   AD         0E      02  EF 02A2D         EXTZV   #2, #14, TEMP_BUF1+6, R0                 ; 3999
  60    AE             0E         00      50  F0 02A33         INSV    R0, #0, #14, TEMP_BUF2                   ;
                       54         52      54  C3 02A39         SUBL3   FLOAT_SCALE, R2, FLOAT_SCALE             ; 4000
                                          08  15 02A3D 567$:   BLEQ    568$                                     ; 4001
                           60     AE      02  C6 02A3F         DIVL2   #2, TEMP_BUF2                            ; 4003
                                          54  D7 02A43         DECL    FLOAT_SCALE                              ; 4004
                                          F6  11 02A45         BRB     567$                                     ; 4001
                                  05      56  E9 02A47 568$:   BLBC    SIGN, 569$                               ; 4006
```

```
                                                60 AE      60 AE  CE 02A4A          MNEGL   TEMP_BUF2, TEMP_BUF2                    ;
                                                34 BE      60 AE  D0 02A4F 569$:    MOVL    TEMP_BUF2, @OUTPUT                      ; 4007
                                                           19     11 02A54 570$:    BRB     574$                                    ; 3788
                                                      52   F5 AD  3C 02A56 571$:    MOVZWL  DST_INFO+5, R2                          ; 4018
                                                      50         01 CE 02A5A       MNEGL   #1, I                                   ; 4020
                                                           0C    11 02A5D          BRB     573$                                    ;
                                                51 B0 AD      01   50 EF 02A5F 572$:    EXTZV   I, #1, INTMED_DATA, R1          ;
                                                34 BE         01   50 F0 02A65          INSV    R1, I, #1, @OUTPUT              ;
                                                              F0   52 F2 02A6B 573$:    AOBLSS  R2, I, 572$                     ; 4018
                                                           0566    31 02A6F 574$:    BRW     649$                                    ; 2187
                                                              16   6B 8F 02A72 575$:    CASEB   (R11), #4, #22                  ; 4030
                                                         002E  002E   0037    02A76 576$:    .WORD   578$-576$,-                     ;
                                                         002E  002E   00D1    02A7E          584$-576$,-                     ;
                                                         002E  002E   00D1    02A86          577$-576$,-                     ;
                                                         002E  002E   002E    02A8E          577$-576$,-                     ;
                                                         002E  002E   002E    02A96          577$-576$,-                     ;
                                                         002E  0120   002E    02A9E          584$-576$,-                     ;
                                                                              02AA6          577$-576$,-                     ;
                                                                              02AAE          577$-576$,-                     ;
                                                                              02AB6          577$-576$,-                     ;
                                                                              02ABE          577$-576$,-                     ;
                                                                              02AC6          577$-576$,-                     ;
                                                                              02ACE          577$-576$,-                     ;
                                                                              02AD6          577$-576$,-                     ;
                                                                              02ADE          577$-576$,-                     ;
                                                                              02AE6          577$-576$,-                     ;
                                                                              02AEE          577$-576$,-                     ;
                                                                              02AF6          577$-576$,-                     ;
                                                                              02AFE          577$-576$,-                     ;
                                                                              02B06          577$-576$,-                     ;
                                                                              02B0E          577$-576$,-                     ;
                                                                              02B16          577$-576$,-                     ;
                                                                              02B1E          588$-576$                       ;
                                                00000000'  EF   9F  02AA4 577$:    PUSHAB  P.AKH                                   ; 4140
                                                      051C 31   02AAA          BRW     647$                                    ;
                                                58 AE    FD   AD  B0 02AAD 578$:    MOVW    SRC_INFO+5, CLASS_S_DESC                ; 4035
                                                5C AE    F9   AD  D0 02AB2          MOVL    SRC_INFO+1, CLASS_S_DESC+4              ; 4036
                                                   7E    55   8F  9A 02AB7          MOVZBL  #85, -(SP)                              ; 4038
                                                   7E    34   AE  CE 02ABB          MNEGL   SCALE, -(SP)                            ; 4037
                                                               7E  D4 02ABF          CLRL    -(SP)                                   ;
                                                         FF7C  CD  9F 02AC1          PUSHAB  TEMP_BUF1                               ;
                                                           68  AE  9F 02AC5          PUSHAB  CLASS_S_DESC                            ;
                                                00000000G  00      05 FB 02AC8       CALLS   #5, OTS$CVT_T_D                         ;
                                                           6E      50 D0 02ACF       MOVL    R0, STATUS                              ;
                                                           15      6E E8 02AD2       BLBS    STATUS, 579$                            ; 4039
                                                                00000000G  00 DD 02AD5       PUSHL   DBG$GL_OPCODE_NAME              ;
                                                                           01 DD 02ADB       PUSHL   #1                              ;
                                                                00028298   8F DD 02ADD       PUSHL   #164504                         ;
                                                00000000G  00              03 FB 02AE3       CALLS   #3, LIB$SIGNAL                  ;
                                                         FF7D  CD  95 02AEA 579$:    TSTB    TEMP_BUF1+1                             ; 4040
                                                               15  18 02AEE          BGEQ    580$                                    ;
                                                                00000000G  00 DD 02AF0       PUSHL   DBG$GL_OPCODE_NAME              ;
                                                                           01 DD 02AF6       PUSHL   #1                              ;
                                                                00028EF0   8F DD 02AF8       PUSHL   #167664                         ;
                                                00000000G  00              03 FB 02AFE       CALLS   #3, LIB$SIGNAL                  ;
                                                      10 BE    FF7C  CD  71 02B05 580$:    CMPD    TEMP_BUF1, @LRGST_D_LU          ; 4041
```

```
                                                50  DC  02B0B                MOVPSL  R0
        50              50              02      02  EF  02B0D                EXTZV   #2, #2, R0, R0                          ;
                                        15      50  F4  02B12                SOBGEQ  R0, 581$                                ;
                                00000000G       00  DD  02B15                PUSHL   DBG$GL_OPCODE_NAME                      ;
                                                01  DD  02B1B                PUSHL   #1                                      ;
                                000286A3        8F  DD  02B1D                PUSHL   #165539                                 ;
                        00000000G   00          03  FB  02B23                CALLS   #3, LIB$SIGNAL                          ;
                                        00E0    8F  B9  02B2A  581$:         BICPSW  #224                                    ; 4042
                                    09    0C    AE  E9  02B2E                BLBC    CVT_ROUND_FLAG, 582$                    ; 4043
                                34  BE      FF7C  CD  6B  02B32                CVTRDL  TEMP_BUF1, @OUTPUT                      ; 4045
                                                0006  31  02B38                BRW     583$                                    ;
                                34  BE      FF7C  CD  6A  02B3B  582$:         CVTDL   TEMP_BUF1, @OUTPUT                      ; 4047
                                        00E0    8F  B8  02B41  583$:         BISPSW  #224                                    ; 4049
                                                4C  11  02B45                BRB     587$                                    ; 4030
                                58  AE      FD    AD  B0  02B47  584$:         MOVW    SRC_INFO+5, CLASS_S_DESC                ; 4054
                                5C  AE      F9    AD  D0  02B4C                MOVL    SRC_INFO+1, CLASS_S_DESC+4              ; 4055
                                7E          55    8F  9A  02B51                MOVZBL  #85, -(SP)                              ; 4057
                                7E          34    AE  CE  02B55                MNEGL   SCALE, -(SP)                            ; 4056
                                                  7E  D4  02B59                CLRL    -(SP)                                   ;
                                        FF7C    CD  9F  02B5B                PUSHAB  TEMP_BUF1                               ;
                                          68    AE  9F  02B5F                PUSHAB  CLASS_S_DESC                            ;
                        00000000G   00          05  FB  02B62                CALLS   #5, OTS$CVT_T_H                         ;
                                    6E          50  D0  02B69                MOVL    R0, STATUS                              ;
                                    15          6E  E8  02B6C                BLBS    STATUS, 585$                            ; 4058
                                00000000G       00  DD  02B6F                PUSHL   DBG$GL_OPCODE_NAME                      ;
                                                01  DD  02B75                PUSHL   #1                                      ;
                                00028298        8F  DD  02B77                PUSHL   #164504                                 ;
                        00000000G   00          03  FB  02B7D                CALLS   #3, LIB$SIGNAL                          ;
                                    50      FF7C    CD  9E  02B84  585$:         MOVAB   TEMP_BUF1, R0                           ; 4059
                                    51        34    AE  D0  02B89  586$:         MOVL    OUTPUT, R1                              ;
                                00000000G       00  16  02B8D                JSB     DBG$CVT_CVTRHQ_R1                       ;
                                                0442  31  02B93  587$:         BRW     649$                                    ;
                                                  58  D4  02B96  588$:         CLRL    SIGN_FLAG                               ; 4063
                                52        34    AE  D0  02B98                MOVL    OUTPUT, R2                              ; 4073
                                          08    A2  7C  02B9C                CLRQ    8(R2)                                   ; 4074
                                          04    A2  D4  02B9F                CLRL    4(R2)                                   ; 4075
                                62      F9      BD  9A  02BA2                MOVZBL  @SRC_INFO+1, (R2)                       ; 4076
                                62              30  C2  02BA6                SUBL2   #48, (R2)                               ;
                                                02  18  02BA9                BGEQ    589$                                    ; 4081
                                                05  11  02BAB                BRB     590$                                    ;
                                09              62  D1  02BAD  589$:         CMPL    (R2), #9                                ;
                                                2F  15  02BB0                BLEQ    592$                                    ;
                        FFFFFFFD    8F          62  D1  02BB2  590$:         CMPL    (R2), #-3                               ; 4083
                                                12  12  02BB9                BNEQ    591$                                    ;
                                        F9      AD  D6  02BBB                INCL    SRC_INFO+1                              ; 4086
                                        FD      AD  B7  02BBE                DECW    SRC_INFO+5                              ; 4087
                                62      F9      BD  9A  02BC1                MOVZBL  @SRC_INFO+1, (R2)                       ; 4088
                                62              30  C2  02BC5                SUBL2   #48, (R2)                               ;
                                58              01  D0  02BC8                MOVL    #1, SIGN_FLAG                           ; 4090
                                                14  11  02BCB                BRB     592$                                    ; 4083
                                        F9      AD  DD  02BCD  591$:         PUSHL   SRC_INFO+1                              ; 4093
                                                01  DD  02BD0                PUSHL   #1                                      ;
                                                02  DD  02BD2                PUSHL   #2                                      ;
                                00028AAA        8F  DD  02BD4                PUSHL   #166570                                 ;
                        00000000G   00          04  FB  02BDA                CALLS   #4, LIB$SIGNAL                          ;
                                    56      FD      AD  3C  02BE1  592$:         MOVZWL  SRC_INFO+5, R6                          ; 4100
                                                53  D4  02BE5                CLRL    CURRENT_CHAR_NUM                        ; 4103
```

```
                                32  11 02BE7                BRB     597$                                    ;
                            50  52  D0 02BE9 593$:          MOVL    R2, R0                                  ; 4102
                00000000G   00  16 02BEC                   JSB     DBG$CVT_SCALE_OU_UP_BY_10_R1            ;
                            54  F9 BD43  9A 02BF2           MOVZBL  @SRC_INFO+1[CURRENT_CHAR_NUM], -        ; 4104
                                                                    CURRENT_CHARACTER                       ;
                            54      30  C2 02BF7           SUBL2   #48, CURRENT_CHARACTER                  ;
                                    02  18 02BFA           BGEQ    594$                                    ; 4110
                                    05  11 02BFC           BRB     595$                                    ;
                            09      54  D1 02BFE 594$:     CMPL    CURRENT_CHARACTER, #9                   ; 4111
                                    15  15 02C01           BLEQ    596$                                    ;
                                F9 BD43  9F 02C03 595$:     PUSHAB  @SRC_INFO+1[CURRENT_CHAR_NUM]           ; 4113
                                    01  DD 02C07           PUSHL   #1                                      ;
                                    02  DD 02C09           PUSHL   #2                                      ;
                            00028AAA    8F DD 02C0B         PUSHL   #166570                                 ;
                00000000G   00          04 FB 02C11         CALLS   #4, LIB$SIGNAL                          ;
                            62          54 C0 02C18 596$:   ADDL2   CURRENT_CHARACTER, (R2)                 ; 4116
                        CA  53          56 F2 02C1B 597$:   AOBLSS  R6, CURRENT_CHAR_NUM, 593$              ; 4100
                            2F          58 E9 02C1F         BLBC    SIGN_FLAG, 598$                         ; 4122
                                50      AE 7C 02C22         CLRQ    OCTAWORD_ZERO+8                         ; 4129
                                48      AE 7C 02C25         CLRQ    OCTAWORD_ZERO                           ; 4131
                        62      48  AE      62 C3 02C28     SUBL3   (R2), OCTAWORD_ZERO, (R2)               ; 4133
                                50      4C  AE D0 02C2D     MOVL    OCTAWORD_ZERO, R0                       ;
                                50      04  A2 D9 02C31     SBWC    4(R2), R0                               ;
                                    04  A2      50 D0 02C35 MOVL    R0, 4(R2)                               ;
                                50      50  AE D0 02C39     MOVL    OCTAWORD_ZERO, R0                       ;
                                50      08  A2 D9 02C3D     SBWC    8(R2), R0                               ;
                                    08  A2      50 D0 02C41 MOVL    R0, 8(R2)                               ;
                                50      54  AE D0 02C45     MOVL    OCTAWORD_ZERO, R0                       ;
                                50      0C  A2 D9 02C49     SBWC    12(R2), R0                              ;
                                    0C  A2      50 D0 02C4D MOVL    R0, 12(R2)                              ;
                                            63 11 02C51 598$:   BRB     603$                                ; 4030
                                    58  AE  FD  AD B0 02C53 599$:   MOVW    SRC_INFO+5, CLASS_S_DESC        ; 4145
                                    5C  AE  F9  AD D0 02C58         MOVL    SRC_INFO+1, CLASS_S_DESC+4      ; 4146
                            03              1B      6B 8F 02C5D     CASEB   (R1T), #27, #3                  ; 4147
                0057            0011            0057    0011 02C61 600$:    .WORD   601$-600$,-             ;
                                                                                601$-600$,-                 ;
                                                                                601$-600$,-                 ;
                                                                                604$-600$                   ;
                                        00000000'   EF  9F 02C69        PUSHAB  P.AKI                       ; 4185
                                                0357    31 02C6F        BRW     647$                        ;
                                    7E      55      8F  9A 02C72 601$:  MOVZBL  #85, -(SP)                  ; 4153
                                    7E      34      AE  CE 02C76        MNEGL   SCALE, -(SP)                ; 4152
                                                    7E  D4 02C7A        CLRL    -(SP)                       ;
                                            FF7C    CD  9F 02C7C        PUSHAB  TEMP_BUF1                   ;
                                              68    AE  9F 02C80        PUSHAB  CLASS_S_DESC                ;
                        00000000G   00              05  FB 02C83       CALLS   #5, OTS$CVT_T_G             ;
                                    6E              50  D0 02C8A       MOVL    R0, STATUS                  ;
                                    15              6E  E8 02C8D       BLBS    STATUS, 602$                ; 4154
                                        00000000G   00  DD 02C90       PUSHL   DBG$GL_OPCODE_NAME          ;
                                                    01  DD 02C96       PUSHL   #1                          ;
                                        00028298    8F  DD 02C98       PUSHL   #164504                     ;
                        00000000G   00              03  FB 02C9E       CALLS   #3, LIB$SIGNAL              ;
                                    50      34      AE  D0 02CA5 602$:  MOVL    OUTPUT, R0                  ; 4155
                                    60      FF7C    CD  7D 02CA9        MOVQ    TEMP_BUF1, (R0)             ;
                                    1D              6B  91 02CAE        CMPB    (R11), #29                  ; 4157
                                                    4E  12 02CB1        BNEQ    606$                        ;
                                              08    A0  7C 02CB3        CLRQ    8(R0)                       ; 4164
```

```
                                                 49  11  02CB6 603$:    BRB     606$                                    ;4147
                                              7E 55  8F  9A  02CB8 604$:    MOVZBL  #85, -(SP)                              ;4172
                                              7E 34  AE  CE  02CBC          MNEGL   SCALE, -(SP)                            ;4171
                                                 7E  D4  02CC0          CLRL    -(SP)                                   ;
                                           FF7C  CD  9F  02CC2          PUSHAB  TEMP_BUF1                               ;
                                             68  AE  9F  02CC6          PUSHAB  CLASS_S_DESC                            ;
                                00000000G    00      05  FB  02CC9          CALLS   #5, OTS$CVT_T_H                         ;
                                             6E      50  D0  02CD0          MOVL    R0, STATUS                              ;
                                             15      6E  E8  02CD3          BLBS    STATUS, 605$                            ;4173
                                      00000000G      00  DD  02CD6          PUSHL   DBG$GL_OPCODE_NAME                      ;
                                                     01  DD  02CDC          PUSHL   #1                                      ;
                                      00028298       8F  DD  02CDE          PUSHL   #164504                                 ;
                                00000000G    00      03  FB  02CE4          CALLS   #3, LIB$SIGNAL                          ;
                                             58      34  AE  D0  02CEB 605$:    MOVL    OUTPUT, R8                          ;4174
                                68   FF7C    CD          10  28  02CEF          MOVC3   #16, TEMP_BUF1, (R8)                    ;
                                             1E          6B  91  02CF5          CMPB    (R11), #30                          ;4175
                                                         07  12  02CF8          BNEQ    606$                                ;
                             10      00          6E          00  2C  02CFA          MOVC5   #0, (SP), #0, #16, 16(R8)       ;4181
                                                      10     A8  02CFF                                                      ;
                                                   02D4  31  02D01 606$:    BRW     649$                                    ;2187
                                             02          6B  91  02D04 607$:    CMPB    (R11), #2                           ;4193
                                                         12  13  02D07          BEQL    610$                                ;
                                             0E          6B  91  02D09          CMPB    (R11), #14                          ;
                                                         0D  13  02D0C          BEQL    610$                                ;
                                             25          6B  91  02D0E          CMPB    (R11), #37                          ;
                                                         03  1E  02D11          BGEQU   609$                                ;
                                                   0252  31  02D13 608$:    BRW     643$                                    ;
                                             27          6B  91  02D16 609$:    CMPB    (R11), #39                          ;
                                                         F8  1A  02D19          BGTRU   608$                                ;
                                                      30     AE  D5  02D1B 610$:    TSTL    SCALE                           ;4195
                                                         03  12  02D1E          BNEQ    611$                                ;
                                                   015C  31  02D20          BRW     630$                                    ;
                                58   AE          FD      AD  B0  02D23 611$:    MOVW    SRC_INFO+5, CLASS_S_DESC            ;4198
                                5C   AE          F9      AD  D0  02D28          MOVL    SRC_INFO+1, CLASS_S_DESC+4          ;4199
                                             7E      55  8F  9A  02D2D          MOVZBL  #85, -(SP)                          ;4201
                                             7E      34  AE  CE  02D31          MNEGL   SCALE, -(SP)                        ;4200
                                                         7E  D4  02D35          CLRL    -(SP)                               ;
                                                   FF7C  CD  9F  02D37          PUSHAB  TEMP_BUF1                           ;
                                                     68  AE  9F  02D3B          PUSHAB  CLASS_S_DESC                        ;
                                00000000G    00          05  FB  02D3E          CALLS   #5, OTS$CVT_T_H                     ;
                                             6E          50  D0  02D45          MOVL    R0, STATUS                          ;
                                             03          6E  E8  02D48          BLBS    STATUS, 612$                        ;4202
                                                   0120  31  02D4B          BRW     629$                                    ;
                                58   AE              32      B0  02D4E 612$:    MOVW    #50, CLASS_S_DESC                   ;4205
                                5C   AE          60      AE  9E  02D52          MOVAB   TEMP_BUF2, CLASS_S_DESC+4           ;4206
                                             09      F5  AD  B1  02D57          CMPW    DST_INFO+5, #9                      ;4207
                                                         05  1A  02D5B          BGTRU   613$                                ;
                                             57          21  D0  02D5D          MOVL    #33, DIGITS_IN_FRACT                ;4209
                                                         12  11  02D60          BRB     615$                                ;
                                             50      F5  AD  3C  02D62 613$:    MOVZWL  DST_INFO+5, R0                      ;4211
                                             50          09  C2  02D66          SUBL2   #9, R0                              ;
                                             21          50  D1  02D69          CMPL    R0, #33                             ;
                                                         03  15  02D6C          BLEQ    614$                                ;
                                             50          21  D0  02D6E          MOVL    #33, R0                             ;
                                             57          50  D0  02D71 614$:    MOVL    R0, DIGITS_IN_FRACT                 ;
                                                         04  DD  02D74 615$:    PUSHL   #4                                  ;4212
                                                         7E  7C  02D76          CLRQ    -(SP)                               ;
```

```
                                                57  DD 02D78          PUSHL   DIGITS_IN_FRACT                    ;
                                             68 AE  9F 02D7A          PUSHAB  CLASS_S_DESC                      ;
                                          FF7C CD  9F 02D7D          PUSHAB  TEMP_BUF1                         ;
                                00000000G  00      06  FB 02D81          CALLS   #6, FOR$CVT_H_TE                  ;
                                           6E      50  D0 02D88          MOVL    R0, STATUS                    ;
                                           02      6E  E9 02D8B          BLBC    STATUS, 616$                  ;  4213
                                                   15  11 02D8E          BRB     617$                          ;
                                    00000000'      EF  9F 02D90 616$:     PUSHAB  P.AKJ                         ;
                                                   01  DD 02D96          PUSHL   #1                            ;
                                    00028362       8F  DD 02D98          PUSHL   #164706                       ;
                                00000000G  00      03  FB 02D9E          CALLS   #3, LIB$SIGNAL                ;
                            60  AE             32      20  3B 02DA5 617$:     SKPC    #32, #50, TEMP_BUF2           ;  4214
                                                   02  12 02DAA          BNEQ    618$                          ;
                                                   51  D4 02DAC          CLRL    R1                            ;
                                           50      60  AE  9E 02DAE 618$:     MOVAB   TEMP_BUF2, R0                 ;
                                        5A         51      50  C3 02DB2          SUBL3   R0, R1, BUF_OFFSET    ;
                                        55         32      5A  C3 02DB6          SUBL3   BUF_OFFSET, #50, FINAL_LEN  ;  4215
                                              14   AE      55  D0 02DBA          MOVL    FINAL_LEN, OUTPUT_STR_LEN   ;  4216
                                                   26      6B  91 02DBE          CMPB    (R11), #38            ;  4220
                                                           51  12 02DC1          BNEQ    623$                  ;
                                              58   AE      55  B0 02DC3          MOVW    FINAL_LEN, CLASS_S_DESC     ;  4224
                                        5C    AE    34   AE      01  C1 02DC7          ADDL3   #1, OUTPUT, CLASS_S_DESC+4  ;  4225
                                                   52      58  AE  9E 02DCD          MOVAB   CLASS_S_DESC, R2      ;  4226
                                                   51      60 AE4A  9E 02DD1          MOVAB   TEMP_BUF2[BUF_OFFSET], R1
                                                   50          55  D0 02DD6          MOVL    FINAL_LEN, R0
                                                    00000000G  00  16 02DD9          JSB     LIB$SCOPY_R_DX6
                                                   6E          50  D0 02DDF          MOVL    R0, STATUS
                                                00000000G  8F      6E  D1 02DE2          CMPL    STATUS, #LIB$_STRTRU  ;  4227
                                                           02      13  02DE9          BEQL    619$                  ;
                                                           15      11  02DEB          BRB     620$                  ;
                                                    00000000G      00  DD 02DED 619$:     PUSHL   DBG$GL_OPCODE_NAME    ;
                                                           01      DD  02DF3          PUSHL   #1                    ;
                                                    000286AB       8F  DD 02DF5          PUSHL   #165547               ;
                                                00000000G  00      03  FB 02DFB          CALLS   #3, LIB$SIGNAL        ;
                                                           09      6E  E8 02E02 620$:     BLBS    STATUS, 622$          ;  4228
                                                           6E      DD  02E05 621$:     PUSHL   STATUS                ;
                                                00000000G  00      01  FB 02E07          CALLS   #1, LIB$SIGNAL        ;
                                                     34    BE      55  90 02E0E 622$:     MOVB    FINAL_LEN, @OUTPUT    ;  4229
                                                           58      11  02E12          BRB     628$                  ;  4218
                                                           27      6B  91 02E14 623$:     CMPB    (R11), #39            ;  4232
                                                                   03  13 02E17          BEQL    624$                  ;
                                                                E8DE   31 02E19          BRW     325$                  ;
                                                        58      AE      55  B0 02E1C 624$:     MOVW    FINAL_LEN, CLASS_S_DESC     ;  4236
                                                        5C      AE  34  AE  D0 02E20          MOVL    OUTPUT, CLASS_S_DESC+4      ;  4237
                                                                52  58  AE  9E 02E25          MOVAB   CLASS_S_DESC, R2      ;  4238
                                                                51  60 AE4A  9E 02E29          MOVAB   TEMP_BUF2[BUF_OFFSET], R1
                                                                50      55  D0 02E2E          MOVL    FINAL_LEN, R0
                                                                 00000000G  00  16 02E31          JSB     LIB$SCOPY_R_DX6
                                                                6E          50  D0 02E37          MOVL    R0, STATUS
                                                             00000000G  8F      6E  D1 02E3A          CMPL    STATUS, #LIB$_STRTRU  ;  4239
                                                                        12      15  02E41          BNEQ    625$                  ;
                                                                 00000000G      00  DD 02E43          PUSHL   DBG$GL_OPCODE_NAME    ;
                                                                        01      DD  02E49          PUSHL   #1                    ;
                                                                 000286AB       8F  DD 02E4B          PUSHL   #165547               ;
                                                             00000000G  00      03  FB 02E51          CALLS   #3, LIB$SIGNAL        ;
                                                                        09      6E  E8 02E58 625$:     BLBS    STATUS, 627$          ;  4240
                                                                        6E      DD  02E5B 626$:     PUSHL   STATUS                ;
```

```
                00000000G  00          01  FB  02E5D                CALLS   #1, LIB$SIGNAL                          ;
             50            55      34  AE  C1  02E64  627$:        ADDL3   OUTPUT, FINAL_LEN, R0                  ; 4241
                               01      A0  94  02E69                CLRB    1(R0)                                   ;
                                       64  11  02E6C  628$:        BRB     634$                                    ; 4218
                                00000000G  00  DD  02E6E  629$:     PUSHL   DBG$GL_OPCODE_NAME                      ; 4253
                                           01  DD  02E74            PUSHL   #1                                      ;
                                00028298   8F  DD  02E76            PUSHL   #164504                                 ;
                                         0152  31  02E7C            BRW     648$                                    ;
                          14    AE              6A  3C  02E7F  630$:  MOVZWL  (R10), OUTPUT_STR_LEN                  ; 4257
                          26                    6B  91  02E83        CMPB    (R11), #38                              ; 4260
                                                4D  12  02E86        BNEQ    635$                                    ;
                          58    AE              6A  B0  02E88        MOVW    (R10), CLASS_S_DESC                     ; 4264
                5C  AE    34    AE              01  C1  02E8C        ADDL3   #1, OUTPUT, CLASS_S_DESC+4              ; 4265
                                51      58    AE  9E  02E92        MOVAB   CLASS_S_DESC, R1                        ; 4266
                                50              5A  D0  02E96        MOVL    R10, R0                                 ;
                                00000000G       00  16  02E99        JSB     LIB$SCOPY_DXDX6                         ;
                                           6E   50  D0  02E9F        MOVL    R0, STATUS                              ;
                00000000G  8F              6E   D1  02EA2        CMPL    STATUS, #LIB$_STRTRU                    ; 4267
                                                02  13  02EA9        BEQL    631$                                    ;
                                                15  11  02EAB        BRB     632$                                    ;
                                00000000G  00   DD  02EAD  631$:     PUSHL   DBG$GL_OPCODE_NAME                      ;
                                           01   DD  02EB3            PUSHL   #1                                      ;
                                000286AB   8F   DD  02EB5            PUSHL   #165547                                 ;
                00000000G  00              03   FB  02EBB            CALLS   #3, LIB$SIGNAL                          ;
                           09              6E   E8  02EC2  632$:     BLBS    STATUS, 633$                            ; 4268
                                           6E   DD  02EC5            PUSHL   STATUS                                  ;
                00000000G  00              01   FB  02EC7            CALLS   #1, LIB$SIGNAL                          ;
                      34   BE              6A   90  02ECE  633$:     MOVB    (R10), @OUTPUT                          ; 4269
                                         0091   31  02ED2  634$:     BRW     642$                                    ; 4258
                           27              6B   91  02ED5  635$:     CMPB    (R11), #39                              ; 4272
                                           51   12  02ED8            BNEQ    639$                                    ;
                          58    AE              6A  B0  02EDA        MOVW    (R10), CLASS_S_DESC                     ; 4276
                5C        AE        34    AE    D0  02EDE        MOVL    OUTPUT, CLASS_S_DESC+4                  ; 4277
                                51      58    AE  9E  02EE3        MOVAB   CLASS_S_DESC, R1                        ; 4278
                                50              5A  D0  02EE7        MOVL    R10, R0                                 ;
                                00000000G       00  16  02EEA        JSB     LIB$SCOPY_DXDX6                         ;
                                           6E   50  D0  02EF0        MOVL    R0, STATUS                              ;
                00000000G  8F              6E   D1  02EF3        CMPL    STATUS, #LIB$_STRTRU                    ; 4279
                                                02  13  02EFA        BEQL    636$                                    ;
                                                15  11  02EFC        BRB     637$                                    ;
                                00000000G  00   DD  02EFE  636$:     PUSHL   DBG$GL_OPCODE_NAME                      ;
                                           01   DD  02F04            PUSHL   #1                                      ;
                                000286AB   8F   DD  02F06            PUSHL   #165547                                 ;
                00000000G  00              03   FB  02F0C            CALLS   #3, LIB$SIGNAL                          ;
                           09              6E   E8  02F13  637$:     BLBS    STATUS, 638$                            ; 4280
                                           6E   DD  02F16            PUSHL   STATUS                                  ;
                00000000G  00              01   FB  02F18            CALLS   #1, LIB$SIGNAL                          ;
                           50              6A   3C  02F1F  638$:     MOVZWL  (R10), R0                               ; 4281
                                50      34  AE   C0  02F22        ADDL2   OUTPUT, R0                              ;
                                           60   94  02F26            CLRB    (R0)                                    ;
                                         0096   31  02F28            BRW     645$                                    ; 4258
                           51              59   D0  02F28  639$:     MOVL    R9, R1                                  ; 4286
                           50              5A   D0  02F2E            MOVL    R10, R0                                 ;
                                00000000G  00   16  02F31            JSB     LIB$SCOPY_DXDX6                         ;
                                           6E   50  D0  02F37        MOVL    R0, STATUS                              ;
                00000000G  8F              6E   D1  02F3A        CMPL    STATUS, #LIB$_STRTRU                    ; 4287
                                                02  13  02F41        BEQL    640$                                    ;
```

```
                                           15  11 02F43        BRB     641$                                   ;
                              00000000G    00  DD 02F45 640$:  PUSHL   DBG$GL_OPCODE_NAME                     ;
                                           01  DD 02F4B        PUSHL   #1                                     ;
                              000286AB     8F  DD 02F4D        PUSHL   #165547                                ;
                   00000000G  00           03  FB 02F53        CALLS   #3, LIB$SIGNAL                         ;
                              7B           6E  E8 02F5A 641$:  BLBS    STATUS, 649$                           ; 4288
                                           6E  DD 02F5D        PUSHL   STATUS                                 ;
                   00000000G  00           01  FB 02F5F        CALLS   #1, LIB$SIGNAL                         ;
                                           70  11 02F66 642$:  BRB     649$                                   ; 4190
                              16           6B  91 02F68 643$:  CMPB    (R11), #22                             ; 4294
                                           56  12 02F6B        BNEQ    646$                                   ;
                   00000000'  EF       FD  AD  90 02F6D        MOVB    SRC_INFO+5, INPUT_STR                  ; 4299
                              50 00000000' EF  9A 02F75        MOVZBL  INPUT_STR, R0                          ; 4300
      00000000'  EF       F9  BD           50  28 02F7C        MOVC3   R0, @SRC_INFO+1, INPUT_STR+1           ;
                                       04  A9  DD 02F85        PUSHL   4(R9)                                  ; 4301
                              00000000'    EF  9F 02F88        PUSHAB  OUTPUT_STR                             ;
                              00000000'    EF  9F 02F8E        PUSHAB  INPUT_STR                              ;
                   00000000G  00           03  FB 02F94        CALLS   #3, DBG$INS_ENCODE                     ;
                              6E           50  D0 02F9B        MOVL    R0, STATUS                             ;
                              09           6E  E8 02F9E        BLBS    STATUS, 644$                           ; 4302
                                           6E  DD 02FA1        PUSHL   STATUS                                 ;
                   00000000G  00           01  FB 02FA3        CALLS   #1, LIB$SIGNAL                         ;
                              69 00000000' EF  9B 02FAA 644$:  MOVZBW  OUTPUT_STR, (R9)                       ; 4303
                              50 00000000' EF  9A 02FB1        MOVZBL  OUTPUT_STR, R0                         ; 4304
             04      B9 00000000'  EF      50  28 02FB8        MOVC3   R0, OUTPUT_STR+1, @4(R9)               ;
                                           15  11 02FC1 645$:  BRB     649$                                   ; 4190
                              00000000'    EF  9F 02FC3 646$:  PUSHAB  P.AKK                                  ; 4308
                                           01  DD 02FC9 647$:  PUSHL   #1                                     ;
                              00028362     8F  DD 02FCB        PUSHL   #164706                                ;
                   00000000G  00           03  FB 02FD1 648$:  CALLS   #3, LIB$SIGNAL                         ;
                              0D       03  A9  91 02FD8 649$:  CMPB    3(R9), #13                             ; 4317
                                           03  13 02FDC        BEQL    650$                                   ;
                                         0090  31 02FDE        BRW     655$                                   ;
                                           53  D4 02FE1 650$:  CLRL    SRC_POS                                ; 4320
                              52       08  A9  D0 02FE3        MOVL    8(R9), DST_POS                         ; 4321
                   29         01           6B  8F 02FE7        CASEB   (R11), #1, #41                         ; 4322
    0088           0088       0088       006B     02FEB 651$:  .WORD   652$-651$,-                            ;
    0088           0088       0088       0088     02FF3                656$-651$,-                            ;
    0088           0088       0088       0088     02FFB                656$-651$,-                            ;
    0088           0088       0088       0088     03003                656$-651$,-                            ;
    0088           0088       0088       0088     0300B                656$-651$,-                            ;
    0088           0088       0088       0088     03013                656$-651$,-                            ;
    0088           0088       0088       0088     0301B                656$-651$,-                            ;
    0088           0088       0088       0088     03023                656$-651$,-                            ;
    0088           0088       006B       0088     0302B                656$-651$,-                            ;
    006B           0088       0088       0088     03033                656$-651$,-                            ;
                              006B       006B     0303B                656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
                                                                       656$-651$,-                            ;
```

```
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              652$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              656$-651$,-                    ;
                                                              652$-651$,-                    ;
                                                              652$-651$,-                    ;
                                                              652$-651$                      ;
                        00000000'  EF  9F 0303F         PUSHAB  P.AKL                        ; 4349
                                   01  DD 03045         PUSHL   #1                           ;
                        00028362   8F  DD 03047         PUSHL   #164706                      ;
              00000000G  00        03  FB 0304D         CALLS   #3, LIB$SIGNAL               ;
                                   3B  11 03054         BRB     659$                         ;
                         54        69  3C 03056 652$:   MOVZWL  (R9), R4                     ; 4330
                                   51  D4 03059         CLRL    I                            ; 4332
                                   10  11 0305B         BRB     654$                         ;
     50        34   BE   01        53  EF 0305D 653$:   EXTZV   SRC_POS, #1, @OUTPUT, R0     ;
 20  BE             01   52        50  F0 03063         INSV    R0, DST_POS, #1, @DESTINATION_PTR ;
                                   52  D6 03069         INCL    DST_POS                      ; 4333
                                   53  D6 0306B         INCL    SRC_POS                      ; 4334
                    EC   51        54  F3 0306D 654$:   AOBLEQ  R4, I, 653$                  ; 4330
                                   1E  11 03071 655$:   BRB     659$                         ; 4322
                         54        69  3C 03073 656$:   MOVZWL  (R9), R4                     ; 4340
                                   51  D4 03076         CLRL    I                            ; 4342
                                 0012  31 03078         BRW     658$                         ;
     50        34   BE   08        53  EF 0307B 657$:   EXTZV   SRC_POS, #8, @OUTPUT, R0     ;
 20  BE             08   52        50  F0 03081         INSV    R0, DST_POS, #8, @DESTINATION_PTR ;
                         52        08  C0 03087         ADDL2   #8, DST_POS                  ; 4343
                         53        08  C0 0308A         ADDL2   #8, SRC_POS                  ; 4344
                    EA   51        54  F3 0308D 658$:   AOBLEQ  R4, I, 657$                  ; 4340
                         02        6C  91 03091 659$:   CMPB    (AP), #2                     ; 4357
                                   05  1B 03094         BLEQU   660$                         ;
                  0C     BC   14   AE  B0 03096         MOVW    OUTPUT_STR_LEN, @OUTLEN      ;
                                   04     0309B 660$:   RET                                  ; 4359
                                 0000     0309C 661$:   .WORD   Save nothing                 ; 1930
                                   7E  D4 0309E         CLRL    -(SP)                        ;
                                   5E  DD 030A0         PUSHL   SP                           ;
                         7E   04   AC  7D 030A2         MOVQ    4(AP), -(SP)                 ;
                  0000V  CF        03  FB 030A6         CALLS   #3, CVT_HANDLER              ;
                                       04 030AB         RET                                  ;

; Routine Size:  12460 bytes,    Routine Base:  DBG$CODE + 02F0
```

```
;  4256          4360  1 ROUTINE CVT_HANDLER (SIG, MECH) =
;  4257          4361  1 !
;  4258          4362  1 ! FUNCTIONAL DESCRIPTION:
;  4259          4363  1 !
;  4260          4364  1 !       This handler will resignal opcode reserved to digital; it
;  4261          4365  1 !       otherwise translates system service signals to debug
;  4262          4366  1 !       error codes and resignals.
;  4263          4367  1 !
;  4264          4368  1 ! FORMAL PARAMETERS:
;  4265          4369  1 !
;  4266          4370  1 !       SIG.rr.r        A counted vector of parameters describing the condition.
;  4267          4371  1 !       MECH.rr.r       A counted vector of parameters from CHF.
;  4268          4372  1 !
;  4269          4373  1 ! IMPLICIT INPUTS:
;  4270          4374  1 !
;  4271          4375  1 !       NONE
;  4272          4376  1 !
;  4273          4377  1 ! IMPLICIT OUTPUTS:
;  4274          4378  1 !
;  4275          4379  1 !       NONE
;  4276          4380  1 !
;  4277          4381  1 ! COMPLETION STATUS: (or ROUTINE VALUE:)
;  4278          4382  1 !
;  4279          4383  1 !       SS$_RESIGNAL when opcode reserved to digital exception.  Any other case
;  4280          4384  1 !       will result in a debug condition being signalled.
;  4281          4385  1 !
;  4282          4386  1 ! SIDE EFFECTS:
;  4283          4387  1 !
;  4284          4388  1 !       NONE
;  4285          4389  1 !
;  4286          4390  2     BEGIN
;  4287          4391  2     MAP
;  4288          4392  2         SIG : REF VECTOR,
;  4289          4393  2         MECH : REF VECTOR;
;  4290          4394  2
;  4291          4395  2
;  4292          4396  2     !Translate error code if this is not an UNWIND, or opcode reserved to digital.
;  4293          4397  2     !Otherwise, signal debug error.
;  4294          4398  2     !
;  4295          4399  2     IF (LIB$MATCH_COND (SIG [1], %REF (SS$_UNWIND), %REF (SS$_OPCDEC))) GTR 0
;  4296          4400  2     THEN
;  4297          4401  2         RETURN (SS$_RESIGNAL);
;  4298          4402  2
;  4299          4403  2
;  4300          4404  2     !Translate all numeric exceptions to debug's facility code.
;  4301          4405  2     !Also, translate SS$_ROPRAND to DBG$_ROPRANDF.
;  4302          4406  2     !
;  4303          4407  2     SELECTONE .SIG[1] OF
;  4304          4408  2         SET
;  4305          4409  2         [SS$_INTOVF]:
;  4306          4410  2             SIGNAL (DBG$_IINTOVF, 1, .DBG$GL_OPCODE_NAME);
;  4307          4411  2         [SS$_DECOVF]:
;  4308          4412  2             SIGNAL (DBG$_DECOVF, 1, .DBG$GL_OPCODE_NAME);
;  4309          4413  2         [SS$_FLTOVF, SS$_FLTOVF_F]:
;  4310          4414  2             SIGNAL (DBG$_FLTOVF, 1, .DBG$GL_OPCODE_NAME);
;  4311          4415  2         [SS$_FLTUND, SS$_FLTUND_F]:
;  4312          4416  3             BEGIN
```

```
;  4313          4417  3              .SAVE_RESULT = 0;
;  4314          4418  3              SIGNAL (DBG$_IFLTUND, 1, .DBG$GL_OPCODE_NAME);
;  4315          4419  2              END;
;  4316          4420  2          [SS$_ROPRAND]:
;  4317          4421  2              BEGIN
;  4318          4422  2              IF .DECIMAL_CONVERT
;  4319          4423  2              THEN
;  4320          4424  2                  SIGNAL (DBG$_DECROPRAND)
;  4321          4425  2              ELSE
;  4322          4426  2                  SIGNAL (DBG$_ROPRANDF, 1, .DBG$GL_OPCODE_NAME);
;  4323          4427  2              END;
;  4324          4428  2          [OTHERWISE]:
;  4325          4429  2              RETURN (SS$_RESIGNAL);
;  4326          4430  2          TES;
;  4327          4431  2
;  4328          4432  2      SETUNWIND();
;  4329          4433  2      RETURN 0;
;  4330          4434  1      END;                                    ! End of CVT_HANDLER


                                001C 00000 CVT_HANDLER:
                                           .WORD   Save R2,R3,R4                    ; 4360
                    54 00000000G   00  9E 00002    MOVAB   LIB$SIGNAL, R4
                    53 00000000G   00  9E 00009    MOVAB   DBG$GL_OPCODE_NAME, R3
                    5E             08  C2 00010    SUBL2   #8, SP
                04  AE      043C   8F  3C 00013    MOVZWL  #1084, 4(SP)             ; 4399
                    04         AE  9F  9F 00019    PUSHAB  4(SP)
                04  AE      0920   8F  3C 0001C    MOVZWL  #2336, 4(SP)
                    04         AE  9F  9F 00022    PUSHAB  4(SP)
                    52         04  AC  D0 00025    MOVL    SIG, R2
                    04         04  A2  9F 00029    PUSHAB  4(R2)
00000000G           00             03  FB 0002C    CALLS   #3, LIB$MATCH_COND
                    52         04  A2  D0 00037    MOVL    4(R2), R2                ; 4407
0000047C            8F             52  D1 0003B    CMPL    R2, #1148                ; 4409
                                   0C  12 00042    BNEQ    1$
                                   63  DD 00044    PUSHL   DBG$GL_OPCODE_NAME       ; 4410
                                   01  DD 00046    PUSHL   #1
                        000286A3   8F  DD 00048    PUSHL   #165539
                                   7C  11 0004E    BRB     9$
000004A4            8F             52  D1 00050 1$:    CMPL    R2, #1188           ; 4411
                                   0C  12 00057    BNEQ    2$
                                   63  DD 00059    PUSHL   DBG$GL_OPCODE_NAME       ; 4412
                                   01  DD 0005B    PUSHL   #1
                        00028A3A   8F  DD 0005D    PUSHL   #166458
                                   67  11 00063    BRB     9$
0000048C            8F             52  D1 00065 2$:    CMPL    R2, #1164           ; 4413
                                   09  13 0006C    BEQL    3$
000004B4            8F             52  D1 0006E    CMPL    R2, #1204
                                   0C  12 00075    BNEQ    4$
                                   63  DD 00077 3$:    PUSHL   DBG$GL_OPCODE_NAME   ; 4414
                                   01  DD 00079    PUSHL   #1
                        00028A02   8F  DD 0007B    PUSHL   #166402
```

```
                                    49  11  00081          BRB     9$                                          ;
                          0000049C  8F      52  D1  00083 4$:     CMPL    R2, #1180                           ; 4415
                                    09  13  0008A          BEQL    5$                                          ;
                          000004C4  8F      52  D1  0008C          CMPL    R2, #1220                           ;
                                    12  12  00093          BNEQ    6$                                          ;
                                00000000' FF  D4  00095 5$:     CLRL    @SAVE_RESULT                        ; 4417
                                    63  DD  00098          PUSHL   DBG$GL_OPCODE_NAME                  ; 4418
                                    01  DD  0009D          PUSHL   #1                                          ;
                                0002869B  8F  DD  0009F          PUSHL   #165531                             ;
                                    25  11  000A5          BRB     9$                                          ;
                          00000454  8F      52  D1  000A7 6$:     CMPL    R2, #1108                           ; 4420
                                    21  12  000AE 7$:     BNEQ    10$                                         ;
                                0B 00000000' EF  E9  000B0          BLBC    DECIMAL_CONVERT, 8$                 ; 4422
                                00028A42  8F  DD  000B7          PUSHL   #166466                             ; 4424
                                64      01  FB  000BD          CALLS   #1, LIB$SIGNAL                      ;
                                    15  11  000C0          BRB     11$                                         ;
                                    63  DD  000C2 8$:     PUSHL   DBG$GL_OPCODE_NAME                  ; 4426
                                    01  DD  000C4          PUSHL   #1                                          ;
                                00028A0A  8F  DD  000C6          PUSHL   #166410                             ;
                                64      03  FB  000CC 9$:     CALLS   #3, LIB$SIGNAL                      ;
                                    06  11  000CF          BRB     11$                                 ; 4407
                                50  0918  8F  3C  000D1 10$:     MOVZWL  #2328, R0                           ; 4429
                                        04  000D6          RET                                                 ;
                                    7E  7C  000D7 11$:     CLRQ    -(SP)                               ; 4432
                          00000000G 00      02  FB  000D9          CALLS   #2, SYS$UNWIND                      ;
                                    50  D4  000E0          CLRL    R0                                  ; 4433
                                        04  000E2          RET                                         ; 4434

; Routine Size:  227 bytes,    Routine Base:  DBG$CODE + 339C
```

```
;  4332        4435  1 ROUTINE FIND_CVT_PATH (SOURCE, DESTINATION, SRC_INFO, DST_INFO, CVT_PATH) =
;  4333        4436  1 !
;  4334        4437  1 ! FUNCTION
;  4335        4438  1 !       This routine is called by DBG$CVT_DX_DX, and has the following four
;  4336        4439  1 !       functions:
;  4337        4440  1 !
;  4338        4441  1 !           a.  It finds any errors concerning the class and data type of the source
;  4339        4442  1 !               and destination descriptors.  These errors can be invalid class,
;  4340        4443  1 !               invalid data type, or invalid combination of a class and data type.
;  4341        4444  1 !               It can also tell which descriptors are supported by this
;  4342        4445  1 !               routine.
;  4343        4446  1 !
;  4344        4447  1 !           b.  It figures out what the conversion path is; ie,
;  4345        4448  1 !               class,dtype --> class,dtype.  These paths are given names such
;  4346        4449  1 !               as K_SMLINT_DEC, which reads "from small integer to decimal"
;  4347        4450  1 !               (categories defined later).
;  4348        4451  1 !
;  4349        4452  1 !           c.  Converts the source data to an intermediate data.  The strategy
;  4350        4453  1 !               used to select the appropriate intermediate data is explained later.
;  4351        4454  1 !
;  4352        4455  1 !           d.  Puts whatever information is needed about the source and destination
;  4353        4456  1 !               descriptor in two structures passed by DBG$CVT_DX_DX.  These
;  4354        4457  1 !               two structures, SRC_INFO and DST_INFO, contain the kind of
;  4355        4458  1 !               information that can only be visible when the class and data
;  4356        4459  1 !               type of the source and destination descriptors are being
;  4357        4460  1 !               manipulated.  These two structures can be expanded to contain
;  4358        4461  1 !               more information as new class, and dta types may require it.
;  4359        4462  1 !
;  4360        4463  1 !       This routine is comprised of a Deterministic Finite Automaton, defined
;  4361        4464  1 !       as a 5 tuple:
;  4362        4465  1 !       States          : There is a state for each CLASS, and CLASS, DATA TYPE
;  4363        4466  1 !                         combination.
;  4364        4467  1 !       Alphabet        : Classes and Data types.
;  4365        4468  1 !       Mappings        : M(CLASS_S , DTYPE_B) := CLASS_S_DTYPE_B
;  4366        4469  1 !                         . . . . . .
;  4367        4470  1 !                         . . . . . .
;  4368        4471  1 !                         M(CLASS_D , DTYPE_W) := error
;  4369        4472  1 !                         . . . . . .
;  4370        4473  1 !                         . . . . . .
;  4371        4474  1 !       Start state     :
;  4372        4475  1 !       Final states    : All possible combinations of CLASS, DTYPE.
;  4373        4476  1 !                         Some of these combinations are allowed, others
;  4374        4477  1 !                         are not. The error combinations are denoted by
;  4375        4478  1 !                         negative numbers as states.
;  4376        4479  1 !
;  4377        4480  1 !
;  4378        4481  1 ! MAINTENANCE OF THIS ROUTINE:
;  4379        4482  1 !
;  4380        4483  1 ! This routine knows about all classes and data types of Appendix C V8.3.
;  4381        4484  1 ! (You may want to update the above line everytime a change is made)
;  4382        4485  1 ! To make an already existing CLASS, DATA TYPE combination a valid one, as
;  4383        4486  1 ! opposed to an error you must:
;  4384        4487  1 !       1. Insert the symbol for that data type in DTYPE_TABLE in place of the
;  4385        4488  1 !          error state.
;  4386        4489  1 !       2. Define a FINAL_STATE for this combination.
;  4387        4490  1 !       3. Give it an action routine.
;  4388        4491  1 !
```

```
;  4389          4492  1 ! To add a new data type you must:
;  4390          4493  1 !       1. Increment K_MAX_DATA_TYPES.
;  4391          4494  1 !       2. Set K_MAX_DTYPE_STA to value of the new data type.
;  4392          4495  1 !       3. Does any of the following need to be changed?
;  4393          4496  1 !               a. K_SMLFINSTA
;  4394          4497  1 !               b. K_LRGFINSTA
;  4395          4498  1 !               c. K_TOP_SD
;  4396          4499  1 !               d. K_BOTTOM_SD
;  4397          4500  1 !       4. Define a new FINAL_STATE.
;  4398          4501  1 !       5. Each category in DTYPE_TABLE must have a new entry for the data type.
;  4399          4502  1 !          Note that the position (starting at 0) of each entry in each category is equivalent
;  4400          4503  1 !          to the data type value.
;  4401          4504  1 !       6. Add the new lable into the action routines CASE statement and
;  4402          4505  1 !          the sub-CASE statements in DBG$CVT_DX_DX will need to be modified to
;  4403          4506  1 !          include this new data type.
;  4404          4507  1 !
;  4405          4508  1 ! To add a new class you must:
;  4406          4509  1 !       1. Increment K_MAX_CLASSES
;  4407          4510  1 !       2. Set K_MAX_CLASS_STA to value of the new class.
;  4408          4511  1 !       3. Increment K_ACTUAL_CLASSES.
;  4409          4512  1 !       4. Make a new K_STATEx_CLASS_y, where x is class value and y is the
;  4410          4513  1 !          symbol of the class.
;  4411          4514  1 !       5. Make a new FINAL_STATE.
;  4412          4515  1 !       6. Add a new category to the STATES structure at the end, with a index
;  4413          4516  1 !          value of one higher than the last category.
;  4414          4517  1 !       7. Make a new entry in CLASS_TABLE.
;  4415          4518  1 !       8. Make a new category in DTYPE_TABLE.
;  4416          4519  1 !       9. Make a new lable in the action routine CASE statement.
;  4417          4520  1 !
;  4418          4521  1 !
;  4419          4522  1 ! CALLING SEQUENCE:
;  4420          4523  1 !       ret_status.wlc.v = FIND_CVT_PATH (SOURCE.rx.dx,
;  4421          4524  1 !                                         DESTINATION.rx.dx,
;  4422          4525  1 !                                         SRC_INFO.wr.r,
;  4423          4526  1 !                                         DST_INFO.wr.r,
;  4424          4527  1 !                                         CVT_PATH.wlu.r)
;  4425          4528  1 !
;  4426          4529  1 ! FORMAL PARAMETERS:
;  4427          4530  1 !       SOURCE          Address of source descriptor passed to DBG$CVT_DX_DX.
;  4428          4531  1 !       DESTINATION     Address of destination descriptor passed to DBG$CVT_DX_DX.
;  4429          4532  1 !       SRC_INFO        Address of a record in DBG$CVT_DX_DX.  Source information goes here.
;  4430          4533  1 !       DST_INFO        Address of a record in DBG$CVT_DX_DX.  Destination info goes here.
;  4431          4534  1 !       CVT_PATH        Address of a longword in DBG$CVT_DX_DX.  This code will determine which
;  4432          4535  1 !                       CASE label is taken in DBG$CVT_DX_DX.
;  4433          4536  1 !
;  4434          4537  1 ! IMPLICIT INPUTS:
;  4435          4538  1 !       NONE
;  4436          4539  1 !
;  4437          4540  1 ! IMPLICIT OUTPUTS:
;  4438          4541  1 !       NONE
;  4439          4542  1 !
;  4440          4543  1 ! COMPLETION STATUS: (or ROUTINE VALUE:)
;  4441          4544  1 !       K_UNSCLAROU             : -1 Unsupported CLASS by routine.
;  4442          4545  1 !       K_UNSDTYROU             : -2 Unsupported DTYPE by routine.
;  4443          4546  1 !       K_UNSDESROU             : -3 Unsupported descriptor by routine.
;  4444          4547  1 !       K_UNSDESSTA             : -4 Unsupported descriptor by standard.
;  4445          4548  1 !       K_UNSCLASTA             : -5 Unsupported CLASS by standard.
```

```
;  4446          4549  1 !       K_UNSDTYSTA             : -6 Unsupported DTYPE by standard.
;  4447          4550  1 !       K_INVNBDS               : -7 Invalid NBDS because array size is greater
;  4448          4551  1 !                                  than WU or dimension is not one.
;  4449          4552  1 !       K_SUPPORTED             :  1 This descriptor is supported.
;  4450          4553  1 !
;  4451          4554  1 ! SIDE EFFECTS:
;  4452          4555  1 !       Caller of DBG$CVT_DX_DX must have LIB$EMULATE as a handler, if the
;  4453          4556  1 !       source or destination descriptor explicitely ask for G, H, O conversions.
;  4454          4557  1 !
;  4455          4558  2     BEGIN
;  4456          4559  2     LOCAL
;  4457          4560  2         STATUS,                                 ! Status of this routine
;  4458          4561  2         STATE,                                  ! State
;  4459          4562  2         CLASS,                                  ! Current CLASS being looked at
;  4460          4563  2         DTYPE,                                  ! Current DTYPE being looked at
;  4461          4564  2         TOKEN,                                  ! The value of each data type supported
;  4462          4565  2         LEFT_CVT : VECTOR [1],                  ! Left side of conversion index.
;  4463          4566  2         RIGHT_CVT : VECTOR [1],                 ! Right side of conversion index.
;  4464          4567  2         LEFT_OR_RIGHT_CVT : REF VECTOR,         ! Left or right side of conversion index.
;  4465          4568  2         SRC_OR_DST_INFO : REF BLOCK [, BYTE],   ! Source or destination info.
;  4466          4569  2         SRC_OR_DST : REF BLOCK [, BYTE],        ! Source or destination.
;  4467          4570  2         TEMP_BUF : BLOCK [K_INTMED_DATA_LENGTH, BYTE];  ! Temporary buffer for reshuffling things.
;  4468          4571  2
;  4469          4572  2     MAP
;  4470          4573  2         SOURCE : REF BLOCK [, BYTE],
;  4471          4574  2         DESTINATION : REF BLOCK [, BYTE],
;  4472          4575  2         SRC_INFO : REF BLOCK [, BYTE] FIELD (SRC_INFO_FIELDS),
;  4473          4576  2         DST_INFO : REF BLOCK [, BYTE] FIELD (DST_INFO_FIELDS);
;  4474          4577  2
;  4475          4578  2
;  4476          4579  2 ! Traverse through the state table twice, once for source, and once for
;  4477          4580  2 ! the destination descriptor.  Each time through, it determines a an
;  4478          4581  2 ! intermediate type; ie, an intermediate type for the source and an
;  4479          4582  2 ! intermediate type for the destination.  Eg.  SMLINT or LRGFLTCMPLX.
;  4480          4583  2 ! The action routines also build SRC_INFO, and DST_INFO, and they
;  4481          4584  2 ! convert the source to its intermediate value.
;  4482          4585  2 ! After determining the intermediate mappings for both the source and
;  4483          4586  2 ! destination descriptors, a formula maps both intermediate states into
;  4484          4587  2 ! one final state, eg. K_SMLINT_LRGFLTCMPLX.  This final result is used
;  4485          4588  2 ! as the main CASE index in DBG$CVT_DX_DX.
;  4486          4589  2 !
;  4487          4590  2 ! The loop goes from 0 to 3:  once for source, once for destination; if it makes
;  4488          4591  2 ! it to .TURN EQL 2, then it exits the loop with a successful status.
;  4489          4592  2 ! If the state table indicates an error (eg. invalid dtype-class combination),
;  4490          4593  2 ! or an error is detected in an action routine (eg. size of array cannot fit in WU),
;  4491          4594  2 ! then the routine exits the loop with an error code.
;  4492          4595  2 !
;  4493          4596  3     BEGIN
;  4494          4597  4     STATUS = (INCRU TURN FROM 0 TO 3 DO
;  4495          4598  5         BEGIN
;  4496          4599  5
;  4497          4600  5
;  4498          4601  5         ! Determine CLASS and DTYPE of this go around, also set up LEFT_OR_RIGHT_CVT,
;  4499          4602  5         ! and SRC_OR_DST, and SRC_OR_DST_INFO.
;  4500          4603  5         ! If this is the third time through this loop, we are finished.
;  4501          4604  5         !
;  4502          4605  5         CASE .TURN FROM 0 TO 2 OF
```

```
;  4503          4606  5              SET
;  4504          4607  5              [0]:
;  4505          4608  6                  BEGIN
;  4506          4609  6                  CLASS = .SOURCE [DSC$B_CLASS];
;  4507          4610  6                  DTYPE = .SOURCE [DSC$B_DTYPE];
;  4508          4611  6                  SRC_OR_DST = .SOURCE;
;  4509          4612  6                  SRC_OR_DST_INFO = .SRC_INFO;
;  4510          4613  6                  LEFT_OR_RIGHT_CVT = LEFT_CVT;
;  4511          4614  5                  END;
;  4512          4615  5              [1]:
;  4513          4616  6                  BEGIN
;  4514          4617  6                  CLASS = .DESTINATION [DSC$B_CLASS];
;  4515          4618  6                  DTYPE = .DESTINATION [DSC$B_DTYPE];
;  4516          4619  6                  SRC_OR_DST = .DESTINATION;
;  4517          4620  6                  SRC_OR_DST_INFO = .DST_INFO;
;  4518          4621  6                  LEFT_OR_RIGHT_CVT = RIGHT_CVT;
;  4519          4622  5                  END;
;  4520          4623  5              [2]:
;  4521          4624  5                  EXITLOOP K_SUPPORTED;
;  4522          4625  5              TES;
;  4523          4626  5
;  4524          4627  5          ! Filter out the out-of-range CLASS and DTYPE.
;  4525          4628  5          !
;  4526          4629  5          IF .CLASS GTRU K_MAX_CLASS_STA THEN EXITLOOP K_UNSCLASTA;
;  4527          4630  5          IF .DTYPE GTRU K_MAX_DTYPE_STA THEN EXITLOOP K_UNSDTYSTA;
;  4528          4631  5
;  4529          4632  5          ! Crank up the finite state machine. start looking in the start state.
;  4530          4633  5          !
;  4531          4634  5          STATE = .CLASS_TABLE [.CLASS];
;  4532          4635  5
;  4533          4636  5          ! Action code for each state that results from the start state.
;  4534          4637  5          !
;  4535          4638  5          CASE .STATE FROM K_MSTNEGERR TO K_LRGCLSSUP OF
;  4536          4639  5              SET
;  4537          4640  5              [K_INVNBDS TO K_UNSCLAROU] :
;  4538          4641  5
;  4539          4642  5                  ! Exit the INCR with the error resulting from the
;  4540          4643  5                  ! start state.
;  4541          4644  5                  !
;  4542          4645  5                  EXITLOOP .STATE;
;  4543          4646  5              [K_SMLCLSSUP TO K_LRGCLSSUP] :
;  4544          4647  5                  BEGIN
;  4545          4648  5
;  4546          4649  5                  ! This is a final state, but some constants need to be
;  4547          4650  5                  ! applied to it yet.  This is just a data type, or a
;  4548          4651  6                  ! negative number if error.
;  4549          4652  6                  !
;  4550          4653  6                  TOKEN = .DTYPE_TABLE [.STATE, .DTYPE];
;  4551          4654  6
;  4552          4655  6                  ! Exit INCR with the error resulting in a final state.
;  4553          4656  6                  !
;  4554          4657  6
;  4555          4658  6
;  4556          4659  6
;  4557          4660  6
;  4558          4661  6
;  4559          4662  6
```

```
;  4560          4663  6                IF .TOKEN LSS 0 THEN EXITLOOP .TOKEN;
;  4561          4664  6
;  4562          4665  6
;  4563          4666  6                ! Find the final state.
;  4564          4667  6                !
;  4565          4668  6                STATE = FINAL_STATE (.STATE, .TOKEN);
;  4566          4669  5                END;
;  4567          4670  5            [INRANGE, OUTRANGE] :
;  4568          4671  5                $DBG_ERROR ('DBGCVTDX\FIND_CVT_PATH:  invalid state');
;  4569          4672  5            TES;
;  4570          4673  5
;  4571          4674  5        ! This CASE statement contains the action code for each final state other than
;  4572          4675  5        ! the error states.
;  4573          4676  5        ! The caller of this routine has set up the pointer and length of SRC_INFO
;  4574          4677  5        ! to be the intermediate data area (INTMED_DATA); in the CASE below we change
;  4575          4678  5        ! the pointer and length if needed (e.g. any NBDS), otherwise we never
;  4576          4679  5        ! touch it.
;  4577          4680  5        ! If .TURN is 0 then we are processing the left side of the conversion, when
;  4578          4681  5        ! it is 1 we are processing the right side of the conversion. In other words,
;  4579          4682  5        ! if .TURN is 0 we are looking at the CLASS, DATA TYPE of source; if .TURN
;  4580          4683  5        ! is 1 we are looking at CLASS, DATA TYPE of destination.
;  4581          4684  5        ! These action codes determine which category (e.g. K_SMLINT or K_DEC as
;  4582          4685  5        ! described in DBG$CVT_DX_DX documentation) the source or destination data type
;  4583          4686  5        ! falls into.  They also convert the source data type to an intermediate
;  4584          4687  5        ! data type.  For more detail refer to the functional description of
;  4585          4688  5        ! DBG$CVT_DX_DX.
;  4586          4689  5        !
;  4587          4690  5        CASE .STATE FROM K_SMLFINSTA TO K_LRGFINSTA OF
;  4588          4691  5            SET
;  4589          4692  5
;  4590          4693  5            [K_S_BU, K_SD_BU, K_UBS_BU]:
;  4591          4694  6                BEGIN
;  4592          4695  6                .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4593          4696  6                IF .STATE EQL K_SD_BU THEN
;  4594          4697  7                    BEGIN
;  4595          4698  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4596          4699  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4597          4700  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4598          4701  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4599          4702  6                    END;
;  4600          4703  6                IF .TURN EQL 0
;  4601          4704  6                THEN
;  4602          4705  6                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 8, 0;,
;  4603          4706  6                        BYTE];
;  4604          4707  5                END;
;  4605          4708  5
;  4606          4709  5            [K_S_WU, K_SD_WU, K_UBS_WU]:
;  4607          4710  6                BEGIN
;  4608          4711  6                .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4609          4712  6                IF .STATE EQL K_SD_WU THEN
;  4610          4713  7                    BEGIN
;  4611          4714  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4612          4715  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4613          4716  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4614          4717  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4615          4718  6                    END;
;  4616          4719  6                IF .TURN EQL 0
```

```
;  4617          4720  6                    THEN
;  4618          4721  6                        .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 16, 0;,
;  4619          4722  6                            BYTE];
;  4620          4723  5                    END;
;  4621          4724  5
;  4622          4725  5                [K_S_LU, K_SD_LU, K_UBS_LU]:
;  4623          4726  6                    BEGIN
;  4624          4727  6                    .LEFT_OR_RIGHT_CVT = K_LRGINT;
;  4625          4728  6                    IF .STATE EQL K_SD_LU THEN
;  4626          4729  7                        BEGIN
;  4627          4730  7                        SRC_OR_DST_INFO [M_SCALE] =
;  4628          4731  7                            .SRC_OR_DST [DSC$B_SCALE];
;  4629          4732  7                        SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4630          4733  7                            .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4631          4734  6                        END;
;  4632          4735  6                    IF .TURN EQL 0
;  4633          4736  6                    THEN
;  4634          4737  6                        .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;,
;  4635          4738  6                            BYTE];
;  4636          4739  5                    END;
;  4637          4740  5
;  4638          4741  5                [K_S_B, K_SD_B, K_UBS_B]:
;  4639          4742  6                    BEGIN
;  4640          4743  6                    .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4641          4744  6                    IF .STATE EQL K_SD_B THEN
;  4642          4745  7                        BEGIN
;  4643          4746  7                        SRC_OR_DST_INFO [M_SCALE] =
;  4644          4747  7                            .SRC_OR_DST [DSC$B_SCALE];
;  4645          4748  7                        SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4646          4749  7                            .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4647          4750  6                        END;
;  4648          4751  6                    IF .TURN EQL 0
;  4649          4752  6                    THEN
;  4650          4753  6                        .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 8, 1;,
;  4651          4754  6                            BYTE];
;  4652          4755  5                    END;
;  4653          4756  5
;  4654          4757  5                [K_S_W, K_SD_W, K_UBS_W]:
;  4655          4758  6                    BEGIN
;  4656          4759  6                    .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4657          4760  6                    IF .STATE EQL K_SD_W THEN
;  4658          4761  7                        BEGIN
;  4659          4762  7                        SRC_OR_DST_INFO [M_SCALE] =
;  4660          4763  7                            .SRC_OR_DST [DSC$B_SCALE];
;  4661          4764  7                        SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4662          4765  7                            .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4663          4766  6                        END;
;  4664          4767  6                    IF .TURN EQL 0
;  4665          4768  6                    THEN
;  4666          4769  6                        .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 16, 1;,
;  4667          4770  6                            BYTE];
;  4668          4771  5                    END;
;  4669          4772  5
;  4670          4773  5                [K_S_L, K_SD_L, K_UBS_L]:
;  4671          4774  6                    BEGIN
;  4672          4775  6                    .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4673          4776  6                    IF .STATE EQL K_SD_L THEN
```

```
;  4674          4777  7                    BEGIN
;  4675          4778  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4676          4779  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4677          4780  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4678          4781  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4679          4782  6                    END;
;  4680          4783  6                IF .TURN EQL 0
;  4681          4784  6                THEN
;  4682          4785  6                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 1;,
;  4683          4786  6                        BYTE];
;  4684          4787  5                END;
;  4685          4788  5
;  4686          4789  5            [K_S_V, K_S_SV, K_S_TF, K_UBS_VU, K_UBS_SVU, K_UBS_TF]:
;  4687          4790  6                BEGIN
;  4688          4791  6                .LEFT_OR_RIGHT_CVT = K_SMLINT;
;  4689          4792  6                SRC_OR_DST_INFO[M_LEN] = .SRC_OR_DST[DSC$W_LENGTH];
;  4690          4793  6                IF .TURN EQL 0
;  4691          4794  6                THEN
;  4692          4795  7                    BEGIN
;  4693          4796  7                    LOCAL
;  4694          4797  7                        BITPOS, SRC_PTR;
;  4695          4798  7
;  4696          4799  7                    IF .SOURCE[DSC$B_CLASS] EQL DSC$K_CLASS_UBS
;  4697          4800  7                    THEN
;  4698          4801  7                        BITPOS = .SOURCE[DSC$L_POS]
;  4699          4802  7                    ELSE
;  4700          4803  7                        BITPOS = 0;
;  4701          4804  7                    SRC_PTR = .SOURCE[DSC$A_BASE];
;  4702          4805  7                    IF .STATE EQL K_S_SV OR .STATE EQL K_UBS_SVU
;  4703          4806  7                    THEN
;  4704          4807  7                        .SRC_INFO[S_POINTER] = .(.SRC_PTR)<.BITPOS, .SOURCE[DSC$W_LENGTH], 1>
;  4705          4808  7                    ELSE
;  4706          4809  8                        BEGIN
;  4707          4810  8                        .SRC_INFO[S_POINTER] = .(.SRC_PTR)<.BITPOS, .SOURCE[DSC$W_LENGTH], 0>;
;  4708          4811  8                        IF .BLOCK [.SRC_INFO[S_POINTER], 0, 31, 1, 0;, BYTE]
;  4709          4812  8                        THEN
;  4710          4813  8                            .LEFT_OR_RIGHT_CVT = K_LRGINT;
;  4711          4814  7                        END;
;  4712          4815  6                    END;
;  4713          4816  5                END;
;  4714          4817  5
;  4715          4818  5            [K_S_Q, K_SD_Q, K_UBS_Q, K_S_QU, K_SD_QU, K_UBS_QU]:
;  4716          4819  6                BEGIN
;  4717          4820  6                .LEFT_OR_RIGHT_CVT = K_LRGINT;
;  4718          4821  6                IF .STATE EQL K_SD_Q OR .STATE EQL K_SD_QU
;  4719          4822  6                THEN
;  4720          4823  7                    BEGIN
;  4721          4824  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4722          4825  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4723          4826  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4724          4827  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4725          4828  6                    END;
;  4726          4829  6                IF .TURN EQL 0
;  4727          4830  6                THEN
;  4728          4831  7                    BEGIN
;  4729          4832  7                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;, BYTE];
;  4730          4833  7                    (.SRC_INFO [S_POINTER] + 4) = .BLOCK [.SOURCE [DSC$A_POINTER] + 4, 0, 0, 32, 0;, BYTE];
```

```
;  4731          4834  7                    IF .BLOCK [.SRC_INFO [S_POINTER], 4, 31, 1, 0;, BYTE]
;  4732          4835  7                    THEN
;  4733          4836  8                        BEGIN
;  4734          4837  8                        .SRC_INFO [S_POINTER] = ..SRC_INFO [S_POINTER] XOR %X'FFFFFFFF';
;  4735          4838  8                        .SRC_INFO [S_POINTER] + 4 = .(.SRC_INFO [S_POINTER] + 4) XOR %X'FFFFFFFF';
;  4736          4839  8                        IF ..SRC_INFO [S_POINTER] EQLU K_LRGST_LU
;  4737          4840  8                        THEN
;  4738          4841  9                            BEGIN
;  4739          4842  9                            .SRC_INFO [S_POINTER] = 0;
;  4740          4843  9                            .SRC_INFO [S_POINTER] + 4 = .(.SRC_INFO [S_POINTER] + 4) + 1;
;  4741          4844  9                            END
;  4742          4845  8                        ELSE
;  4743          4846  8                            .SRC_INFO [S_POINTER] = ..SRC_INFO [S_POINTER] + 1;
;  4744          4847  8                        SRC_INFO [S_SIGN] = 1;
;  4745          4848  7                        END;
;  4746          4849  6                    END;
;  4747          4850  5                END;
;  4748          4851  5
;  4749          4852  5            [K_S_O, K_SD_O, K_UBS_O]:
;  4750          4853  6                BEGIN
;  4751          4854  6                .LEFT_OR_RIGHT_CVT = K_LRGINT;
;  4752          4855  6                IF .STATE EQL K_SD_O THEN
;  4753          4856  7                    BEGIN
;  4754          4857  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4755          4858  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4756          4859  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4757          4860  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4758          4861  6                    END;
;  4759          4862  6                IF .TURN EQL 0
;  4760          4863  6                THEN
;  4761          4864  7                    BEGIN
;  4762          4865  7                    CH$MOVE (16, .SOURCE[DSC$A_POINTER], .SRC_INFO[S_POINTER]);
;  4763          4866  7                    IF .BLOCK [.SRC_INFO [S_POINTER], 12, 31, 1, 0;, BYTE]
;  4764          4867  7                    THEN
;  4765          4868  8                        BEGIN
;  4766          4869  8                        INCR I FROM 0 TO 12 BY 4 DO
;  4767          4870  8                            .SRC_INFO[S_POINTER] + .I = .(.SRC_INFO[S_POINTER] + .I) XOR %X'FFFFFFFF';
;  4768          4871  8                        IF ..SRC_INFO [S_POINTER] EQLU K_LRGST_LU
;  4769          4872  8                        THEN
;  4770          4873  9                            BEGIN
;  4771          4874  9                            .SRC_INFO [S_POINTER] = 0;
;  4772          4875  9                            INCR I FROM 4 TO 12 BY 4 DO
;  4773          4876  9                                .SRC_INFO [S_POINTER] + .I = .(.SRC_INFO [S_POINTER] + .I) + 1;
;  4774          4877  9                            END
;  4775          4878  8                        ELSE
;  4776          4879  8                            .SRC_INFO [S_POINTER] = ..SRC_INFO [S_POINTER] + 1;
;  4777          4880  8                        SRC_INFO [S_SIGN] = 1;
;  4778          4881  7                        END;
;  4779          4882  6                    END;
;  4780          4883  5                END;
;  4781          4884  5
;  4782          4885  5            [K_S_F, K_SD_F, K_UBS_F]:
;  4783          4886  6                BEGIN
;  4784          4887  6                .LEFT_OR_RIGHT_CVT = K_SMLFLT_CMPLX;
;  4785          4888  6                IF .STATE EQL K_SD_F THEN
;  4786          4889  7                    BEGIN
;  4787          4890  7                    SRC_OR_DST_INFO [M_SCALE] =
```

```
;  4788          4891  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4789          4892  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4790          4893  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4791          4894  6                    END;
;  4792          4895  6                IF .TURN EQL 0
;  4793          4896  6                THEN
;  4794          4897  6                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;, BYTE];
;  4795          4898  5                END;
;  4796          4899  5
;  4797          4900  5            [K_S_FC, K_SD_FC, K_UBS_FC]:
;  4798          4901  6                BEGIN
;  4799          4902  6                .LEFT_OR_RIGHT_CVT = K_SMLFLT_CMPLX;
;  4800          4903  6                IF .STATE EQL K_SD_FC THEN
;  4801          4904  7                    BEGIN
;  4802          4905  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4803          4906  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4804          4907  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4805          4908  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4806          4909  6                    END;
;  4807          4910  6                IF .TURN EQL 0
;  4808          4911  6                THEN
;  4809          4912  7                    BEGIN
;  4810          4913  7                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;, BYTE];
;  4811          4914  7
;  4812          4915  7
;  4813          4916  7                    ! Intermediate data type is double complex.
;  4814          4917  7                    !
;  4815          4918  7                    (.SRC_INFO [S_POINTER] + 8) = .BLOCK [.SOURCE [DSC$A_POINTER] + 4, 0, 0, 32, 0;, BYTE];
;  4816          4919  6                    END;
;  4817          4920  5                END;
;  4818          4921  5
;  4819          4922  5            [K_S_D, K_SD_D, K_UBS_D]:
;  4820          4923  6                BEGIN
;  4821          4924  6                .LEFT_OR_RIGHT_CVT = K_SMLFLT_CMPLX;
;  4822          4925  6                IF .STATE EQL K_SD_D THEN
;  4823          4926  7                    BEGIN
;  4824          4927  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4825          4928  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4826          4929  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4827          4930  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4828          4931  6                    END;
;  4829          4932  6                IF .TURN EQL 0
;  4830          4933  6                THEN
;  4831          4934  7                    BEGIN
;  4832          4935  7
;  4833          4936  7
;  4834          4937  7                    ! The intermediate data buffer is init'ialized to zero, so
;  4835          4938  7                    ! don't have to worry about filling i   inary part.
;  4836          4939  7                    ! (Intermediate data type is double com  ex).
;  4837          4940  7                    !
;  4838          4941  7                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;, BYTE];
;  4839          4942  7                    (.SRC_INFO [S_POINTER] + 4) = .BLOCK [.SOURCE [DSC$A_POINTER] + 4, 0, 0, 32, 0;, BYTE];
;  4840          4943  6                    END;
;  4841          4944  5                END;
;  4842          4945  5
;  4843          4946  5            [K_S_DC, K_SD_DC, K_UBS_DC]:
;  4844          4947  6                BEGIN
```

```
;  4845          4948  6                .LEFT_OR_RIGHT_CVT = K_SMLFLT_CMPLX;
;  4846          4949  6                IF .STATE EQL K_SD_D THEN
;  4847          4950  7                    BEGIN
;  4848          4951  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4849          4952  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4850          4953  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4851          4954  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4852          4955  6                    END;
;  4853          4956  6                IF .TURN EQL 0
;  4854          4957  6                THEN
;  4855          4958  6                    CH$MOVE (16, .SOURCE[DSC$A_POINTER], .SRC_INFO[S_POINTER]);
;  4856          4959  5                END;
;  4857          4960  5
;  4858          4961  5            [K_S_G, K_SD_G, K_UBS_G]:
;  4859          4962  6                BEGIN
;  4860          4963  6                .LEFT_OR_RIGHT_CVT = K_LRGFLT_CMPLX;
;  4861          4964  6                IF .STATE EQL K_SD_G THEN
;  4862          4965  7                    BEGIN
;  4863          4966  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4864          4967  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4865          4968  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4866          4969  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4867          4970  6                    END;
;  4868          4971  6                IF .TURN EQL 0
;  4869          4972  6                THEN
;  4870          4973  7                    BEGIN
;  4871          4974  7                    .SRC_INFO [S_POINTER] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 32, 0;, BYTE];
;  4872          4975  7                    (.SRC_INFO [S_POINTER] + 4) = .BLOCK [.SOURCE [DSC$A_POINTER] + 4, 0, 0, 32, 0;, BYTE];
;  4873          4976  6                    END;
;  4874          4977  5                END;
;  4875          4978  5
;  4876          4979  5            [K_S_GC, K_SD_GC, K_UBS_GC]:
;  4877          4980  6                BEGIN
;  4878          4981  6                .LEFT_OR_RIGHT_CVT = K_LRGFLT_CMPLX;
;  4879          4982  6                IF .STATE EQL K_SD_GC THEN
;  4880          4983  7                    BEGIN
;  4881          4984  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4882          4985  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4883          4986  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4884          4987  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4885          4988  6                    END;
;  4886          4989  6                IF .TURN EQL 0
;  4887          4990  6                THEN
;  4888          4991  6                    CH$MOVE (16, .SOURCE[DSC$A_POINTER], .SRC_INFO[S_POINTER]);
;  4889          4992  5                END;
;  4890          4993  5
;  4891          4994  5            [K_S_H, K_SD_H, K_UBS_H]:
;  4892          4995  6                BEGIN
;  4893          4996  6                .LEFT_OR_RIGHT_CVT = K_LRGFLT_CMPLX;
;  4894          4997  6                IF .STATE EQL K_SD_H THEN
;  4895          4998  7                    BEGIN
;  4896          4999  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4897          5000  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4898          5001  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4899          5002  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4900          5003  6                    END;
;  4901          5004  6                IF .TURN EQL 0 THEN CH$MOVE (16, .SOURCE [DSC$A_POINTER], .SRC_INFO [S_POINTER]);
```

```
;  4902          5005  5                END;
;  4903          5006  5
;  4904          5007  5            [K_S_HC, K_SD_HC, K_UBS_HC]:
;  4905          5008  6                BEGIN
;  4906          5009  6                .LEFT_OR_RIGHT_CVT = K_LRGFLT_CMPLX;
;  4907          5010  6                IF .STATE EQL K_SD_HC THEN
;  4908          5011  7                    BEGIN
;  4909          5012  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4910          5013  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4911          5014  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4912          5015  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4913          5016  6                    END;
;  4914          5017  6                IF .TURN EQL 0
;  4915          5018  6                THEN
;  4916          5019  6                    CH$MOVE (32, .SOURCE [DSC$A_POINTER], .SRC_INFO [S_POINTER]);
;  4917          5020  5                END;
;  4918          5021  5
;  4919          5022  5            [K_S_T, K_SD_T, K_UBS_T]:
;  4920          5023  6                BEGIN
;  4921          5024  6                .LEFT_OR_RIGHT_CVT = K_NBDS;
;  4922          5025  6                SRC_OR_DST_INFO [M_LEN] = .SRC_OR_DST [DSC$W_LENGTH];
;  4923          5026  6                IF .STATE EQL K_SD_T THEN
;  4924          5027  7                    BEGIN
;  4925          5028  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4926          5029  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4927          5030  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4928          5031  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4929          5032  6                    END;
;  4930          5033  6                IF .TURN EQL J
;  4931          5034  6                THEN
;  4932          5035  7                    BEGIN
;  4933          5036  7                    SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER];
;  4934          5037  6                    END;
;  4935          5038  5                END;
;  4936          5039  5
;  4937          5040  5            [K_S_NU, K_SD_NU]:
;  4938          5041  6                BEGIN
;  4939          5042  6                .LEFT_OR_RIGHT_CVT = K_DEC;
;  4940          5043  6                IF .STATE EQL K_SD_NU THEN
;  4941          5044  7                    BEGIN
;  4942          5045  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4943          5046  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4944          5047  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4945          5048  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4946          5049  6                    END;
;  4947          5050  6                IF .TURN EQL 0
;  4948          5051  6                THEN
;  4949          5052  7                    BEGIN
;  4950          5053  7                    SRC_INFO [S_LEN] = 31;
;  4951          5054  7                    CVTTP (SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], LIB$AB_CVTTP_U,
;  4952          5055  7                        SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  4953          5056  6                    END;
;  4954          5057  5                END;
;  4955          5058  5
;  4956          5059  5            [K_S_NL, K_SD_NL]:
;  4957          5060  6                BEGIN
;  4958          5061  6                .LEFT_OR_RIGHT_CVT = K_DEC;
```

```
;  4959          5062  6                IF .STATE EQL K_SD_NL THEN
;  4960          5063  7                    BEGIN
;  4961          5064  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4962          5065  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4963          5066  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4964          5067  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4965          5068  6                    END;
;  4966          5069  6                IF .TURN EQL 0
;  4967          5070  6                THEN
;  4968          5071  7                    BEGIN
;  4969          5072  7                    SRC_INFO [S_LEN] = 31;
;  4970          5073  7                    CVTSP (%REF (
;  4971          5074  7                            IF .SOURCE [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .SOURCE [DSC$W_LENGTH] - 1),
;  4972          5075  7                        .SOURCE [DSC$A_POINTER], SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  4973          5076  6                    END;
;  4974          5077  5                END;
;  4975          5078  5
;  4976          5079  5            [K_S_NLO, K_SD_NLO]:
;  4977          5080  6                BEGIN
;  4978          5081  6                .LEFT_OR_RIGHT_CVT = K_DEC;
;  4979          5082  6                IF .STATE EQL K_SD_NLO THEN
;  4980          5083  7                    BEGIN
;  4981          5084  7                    SRC_OR_DST_INFO [M_SCALE] =
;  4982          5085  7                        .SRC_OR_DST [DSC$B_SCALE];
;  4983          5086  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  4984          5087  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  4985          5088  6                    END;
;  4986          5089  6                IF .TURN EQL 0
;  4987          5090  6                THEN
;  4988          5091  7                    BEGIN
;  4989          5092  7                    LOCAL
;  4990          5093  7                        LF_SIGN:  REF VECTOR[, BYTE],
;  4991          5094  7                        RT_SIGN:  REF VECTOR[, BYTE],
;  4992          5095  7                        ZERO_FLAG,
;  4993          5096  7                        SIGN_FLAG,
;  4994          5097  7                        PACK_ZERO: VECTOR [1];
;  4995          5098  7                    PACK_ZERO = UPLIT (%P'+0');
;  4996          5099  7                    SRC_INFO [S_LEN] = 31;
;  4997          5100  7 !                  CH$TRANSLATE (LIB$AB_CVT_O_U, .SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], 0,
;  4998          5101  7 !                      .SOURCE [DSC$W_LENGTH], TEMP_BUF);
;  4999          5102  7 !                  CVTTP (SOURCE [DSC$W_LENGTH], TEMP_BUF, LIB$AB_CVTTP_U, SRC_INFO [S_LEN],
;  5000          5103  7 !                      .SRC_INFO [S_POINTER]);
;  5001          5104  7 !
;  5002          5105  7 ! Orignal code turns negative NLO type into positive NLO.  If the negative
;  5003          5106  7 ! type comes into this piece of code without CH$TRANSLATE gives Reserved
;  5004          5107  7 ! Operand fault.  What I did here is to move the left overpunched sign
;  5005          5108  7 ! to the right overpunched sign then performs the conversion.  After the
;  5006          5109  7 ! conversion, put the sign back to where it belonged.
;  5007          5110  7 !
;  5008          5111  7                    RT_SIGN = .SOURCE [DSC$A_POINTER] + .SOURCE [DSC$W_LENGTH] - 1;
;  5009          5112  7                    LF_SIGN = .SOURCE [DSC$A_POINTER];
;  5010          5113  7                    ZERO_FLAG = FALSE;
;  5011          5114  7                    SELECTONE .LF_SIGN[0] OF
;  5012          5115  7                        SET
;  5013          5116  7
;  5014          5117  7                        ! Positive 1 -- 9
;  5015          5118  7                        !
```

```
;  5016          5119  7                [%X'41' TO %X'49']: SIGN_FLAG = TRUE;
;  5017          5120  7
;  5018          5121  7                ! Negative 1 -- 9
;  5019          5122  7                !
;  5020          5123  7                [%X'4A' TO %X'52']: SIGN_FLAG = FALSE;
;  5021          5124  7
;  5022          5125  7                ! Positive 0
;  5023          5126  7                !
;  5024          5127  7                [%X'7B']:
;  5025          5128  8                    BEGIN
;  5026          5129  8                    SIGN_FLAG = TRUE;
;  5027          5130  8                    ZERO_FLAG = TRUE;
;  5028          5131  8                    LF_SIGN[0] = %X'30';
;  5029          5132  8                    IF .RT_SIGN[0] EQL %X'30'
;  5030          5133  8                    THEN
;  5031          5134  8                        RT_SIGN[0] = %X'7B'
;  5032          5135  8                    ELSE
;  5033          5136  8                        RT_SIGN[0] = .RT_SIGN[0] + %X'10';
;  5034          5137  7                    END;
;  5035          5138  7
;  5036          5139  7                ! Negative 0
;  5037          5140  7                !
;  5038          5141  7                [%X'7D']:
;  5039          5142  8                    BEGIN
;  5040          5143  8                    SIGN_FLAG = FALSE;
;  5041          5144  8                    ZERO_FLAG = TRUE;
;  5042          5145  8                    LF_SIGN[0] = %X'30';
;  5043          5146  8                    IF .RT_SIGN[0] EQL %X'30'
;  5044          5147  8                    THEN
;  5045          5148  8                        RT_SIGN[0] = %X'7D'
;  5046          5149  8                    ELSE
;  5047          5150  8                        RT_SIGN[0] = .RT_SIGN[0] + %X'19';
;  5048          5151  7                    END;
;  5049          5152  7
;  5050          5153  7                [OTHERWISE]: $DBG_ERROR('DBGCVTDX\FIND_CVT_PATH');
;  5051          5154  7                TES;
;  5052          5155  7
;  5053          5156  7            IF NOT .ZERO_FLAG
;  5054          5157  7            THEN
;  5055          5158  8                BEGIN
;  5056          5159  8                IF .SIGN_FLAG
;  5057          5160  8                THEN
;  5058          5161  9                    BEGIN
;  5059          5162  9                    LF_SIGN[0] = .LF_SIGN[0] - %X'10';
;  5060          5163  9                    IF .RT_SIGN[0] EQL %X'30'
;  5061          5164  9                    THEN
;  5062          5165  9                        RT_SIGN[0] = %X'7B'
;  5063          5166  9                    ELSE
;  5064          5167  9                        RT_SIGN[0] = .RT_SIGN[0] + %X'10';
;  5065          5168  9                    END
;  5066          5169  9
;  5067          5170  8                ELSE
;  5068          5171  9                    BEGIN
;  5069          5172  9                    LF_SIGN[0] = .LF_SIGN[0] - %X'19';
;  5070          5173  9                    IF .RT_SIGN[0] EQL %X'30'
;  5071          5174  9                    THEN
;  5072          5175  9                        RT_SIGN[0] = %X'7D'
```

```
;  5073          5176  9                    ELSE
;  5074          5177  9                        RT_SIGN[0] = .RT_SIGN[0] + %X'19';
;  5075          5178  8                    END;
;  5076          5179  8
;  5077          5180  7                END;
;  5078          5181  7
;  5079          5182  7            CVTTP (SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], LIB$AB_CVTTP_0,
;  5080          5183  7                   SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  5081          5184  7
;  5082          5185  7            ! Now put the sign back.
;  5083          5186  7            !
;  5084          5187  7            IF .SIGN_FLAG
;  5085          5188  7            THEN
;  5086          5189  8                BEGIN
;  5087          5190  8                IF .RT_SIGN[0] EQL %X'7B'
;  5088          5191  8                THEN
;  5089          5192  8                    RT_SIGN[0] = %X'30'
;  5090          5193  8                ELSE
;  5091          5194  8                    RT_SIGN[0] = .RT_SIGN[0] - %X'10';
;  5092          5195  8
;  5093          5196  8                IF .LF_SIGN[0] EQL %X'30'
;  5094          5197  8                THEN
;  5095          5198  8                    LF_SIGN[0] = %X'7B'
;  5096          5199  8                ELSE
;  5097          5200  8                    LF_SIGN[0] = .LF_SIGN[0] + %X'10';
;  5098          5201  8
;  5099          5202  8                END
;  5100          5203  8
;  5101          5204  7            ELSE
;  5102          5205  8                BEGIN
;  5103          5206  8                IF .RT_SIGN[0] EQL %X'7D'
;  5104          5207  8                THEN
;  5105          5208  8                    RT_SIGN[0] = %X'30'
;  5106          5209  8                ELSE
;  5107          5210  8                    RT_SIGN[0] = .RT_SIGN[0] - %X'19';
;  5108          5211  8
;  5109          5212  8                IF .LF_SIGN[0] EQL %X'30'
;  5110          5213  8                THEN
;  5111          5214  8                    LF_SIGN[0] = %X'7D'
;  5112          5215  8                ELSE
;  5113          5216  8                    LF_SIGN[0] = .LF_SIGN[0] + %X'19';
;  5114          5217  8
;  5115          5218  7                END;
;  5116          5219  7
;  5117          5220  7                IF CMPP (SRC_INFO [S_LEN], .SRC_INFO [S_POINTER], %REF (1), .PACK_ZERO) EQLU 0
;  5118          5221  7                THEN
;  5119          5222  7                    BLOCK [.SRC_INFO [S_POINTER] + .SRC_INFO [S_LEN]/2, 0, 0, 4, 0;, BYTE] = .BLOCK [
;  5120          5223  7                        .LIB$AB_CVTTP_0 + .SOURCE [DSC$A_POINTER], 0, 0, 4, 0;, BYTE];
;  5121          5224  6                END;
;  5122          5225  5            END;
;  5123          5226  5
;  5124          5227  5        [K_S_NR, K_SD_NR]:
;  5125          5228  6            BEGIN
;  5126          5229  6            .LEFT_OR_RIGHT_CVT = K_DEC;
;  5127          5230  6            IF .STATE EQL K_SD_NR THEN
;  5128          5231  7                BEGIN
;  5129          5232  7                SRC_OR_DST_INFO [M_SCALE] =
```

```
;  5130          5233  7                        .SRC_OR_DST [DSC$B_SCALE];
;  5131          5234  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  5132          5235  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  5133          5236  6                    END;
;  5134          5237  6                IF .TURN EQL 0
;  5135          5238  6                THEN
;  5136          5239  7                    BEGIN
;  5137          5240  7                    LOCAL
;  5138          5241  7                        SOU_LEN;
;  5139          5242  7                    SOU_LEN =
;  5140          5243  8                    BEGIN
;  5141          5244  8                    IF .SOURCE [DSC$W_LENGTH] EQL 0 THEN 0 ELSE .SOURCE [DSC$W_LENGTH] - 1
;  5142          5245  7                    END;
;  5143          5246  7                    TEMP_BUF [0, 0, 8, 0] = .BLOCK [.SOURCE [DSC$A_POINTER] + .SOU_LEN, 0, 0, 8, 0;, BYTE];
;  5144          5247  7                    CH$MOVE (.SOU_LEN, .SOURCE [DSC$A_POINTER], TEMP_BUF + 1);
;  5145          5248  7                    SRC_INFO [S_LEN] = 31;
;  5146          5249  7                    CVTSP (SOU_LEN, TEMP_BUF, SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  5147          5250  6                    END;
;  5148          5251  5                END;
;  5149          5252  5
;  5150          5253  5            [K_S_NRO, K_SD_NRO]:
;  5151          5254  6                BEGIN
;  5152          5255  6                .LEFT_OR_RIGHT_CVT = K_DEC;
;  5153          5256  6                IF .STATE EQL K_SD_NRO THEN
;  5154          5257  7                    BEGIN
;  5155          5258  7                    SRC_OR_DST_INFO [M_SCALE] =
;  5156          5259  7                        .SRC_OR_DST [DSC$B_SCALE];
;  5157          5260  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  5158          5261  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  5159          5262  6                    END;
;  5160          5263  6                IF .TURN EQL 0
;  5161          5264  6                THEN
;  5162          5265  7                    BEGIN
;  5163          5266  7                    SRC_INFO [S_LEN] = 31;
;  5164          5267  7                    CVTTP (SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], LIB$AB_CVTTP_0,
;  5165          5268  7                        SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  5166          5269  6                    END;
;  5167          5270  5                END;
;  5168          5271  5
;  5169          5272  5            [K_S_NZ, K_SD_NZ]:
;  5170          5273  6                BEGIN
;  5171          5274  6                .LEFT_OR_RIGHT_CVT = K_DEC;
;  5172          5275  6                IF .STATE EQL K_SD_NZ THEN
;  5173          5276  7                    BEGIN
;  5174          5277  7                    SRC_OR_DST_INFO [M_SCALE] =
;  5175          5278  7                        .SRC_OR_DST [DSC$B_SCALE];
;  5176          5279  7                    SRC_OR_DST_INFO [M_BIN_SCALE] =
;  5177          5280  7                        .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  5178          5281  6                    END;
;  5179          5282  6                IF .TURN EQL 0
;  5180          5283  6                THEN
;  5181          5284  7                    BEGIN
;  5182          5285  7                    SRC_INFO [S_LEN] = 31;
;  5183          5286  7                    CVTTP (SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], LIB$AB_CVTTP_Z,
;  5184          5287  7                        SRC_INFO [S_LEN], .SRC_INFO [S_POINTER]);
;  5185          5288  6                    END;
;  5186          5289  5                END;
```

```
;  5187          5290  5
;  5188          5291  5                [K_S_P, K_SD_P]:
;  5189          5292  6                    BEGIN
;  5190          5293  6                    .LEFT_OR_RIGHT_CVT = K_DEC;
;  5191          5294  6                    IF .STATE EQL K_SD_P THEN
;  5192          5295  7                        BEGIN
;  5193          5296  7                        SRC_OR_DST_INFO [M_SCALE] =
;  5194          5297  7                            .SRC_OR_DST [DSC$B_SCALE];
;  5195          5298  7                        SRC_OR_DST_INFO [M_BIN_SCALE] =
;  5196          5299  7                            .SRC_OR_DST[DSC$V_FL_BINSCALE];
;  5197          5300  6                        END;
;  5198          5301  6                    IF .TURN EQL 0
;  5199          5302  6                    THEN
;  5200          5303  7                        BEGIN
;  5201          5304  7                        CVTPS (SOURCE [DSC$W_LENGTH], .SOURCE [DSC$A_POINTER], %REF (31), TEMP_BUF);
;  5202          5305  7                        CVTSP (%REF (31), TEMP_BUF, %REF (31), .SRC_INFO [S_POINTER]);
;  5203          5306  7                        SRC_INFO [S_LEN] = 31;
;  5204          5307  6                        END;
;  5205          5308  5                    END;
;  5206          5309  5
;  5207          5310  5                [K_S_ZI]:
;  5208          5311  5                    .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5209          5312  5
;  5210          5313  5                [K_D_T]:
;  5211          5314  6                    BEGIN
;  5212          5315  6                    .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5213          5316  6                    SRC_OR_DST_INFO [M_LEN] = .SRC_OR_DST [DSC$W_LENGTH];
;  5214          5317  6                    IF .TURN EQL 0
;  5215          5318  6                    THEN
;  5216          5319  7                        BEGIN
;  5217          5320  7                        SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER];
;  5218          5321  6                        END;
;  5219          5322  5                    END;
;  5220          5323  5
;  5221          5324  5                [K_A_BU, K_A_T, K_NCA_BU, K_NCA_T]:
;  5222          5325  6                    BEGIN
;  5223          5326  6                    .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5224          5327  7                    IF (.SRC_OR_DST [DSC$L_ARSIZE] GTR K_LRGST_WU OR .SRC_OR_DST [DSC$B_DIMCT] NEQ 1 OR
;  5225          5328  7                        .SRC_OR_DST [DSC$W_LENGTH] NEQ 1)
;  5226          5329  6                    THEN
;  5227          5330  6                        EXITLOOP K_INVNBDS;
;  5228          5331  7                    IF (.STATE EQL K_NCA_BU OR .STATE EQL K_NCA_T)
;  5229          5332  6                    THEN
;  5230          5333  7                        BEGIN
;  5231          5334  7                        IF .SRC_OR_DST [DSC$L_S1] NEQ 1 THEN EXITLOOP K_INVNBDS;
;  5232          5335  6                        END;
;  5233          5336  6                    SRC_OR_DST_INFO [M_SCALE] = .SRC_OR_DST [DSC$B_SCALE];
;  5234          5337  6                    SRC_OR_DST_INFO [M_BIN_SCALE] = .SRC_OR_DST [DSC$V_FL_BINSCALE];
;  5235          5338  6                    SRC_OR_DST_INFO [M_LEN] = .SRC_OR_DST [DSC$L_ARSIZE];
;  5236          5339  6                    IF .TURN EQL 0
;  5237          5340  6                    THEN
;  5238          5341  7                        BEGIN
;  5239          5342  7                        SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER];
;  5240          5343  6                        END;
;  5241          5344  5                    END;
;  5242          5345  5
;  5243          5346  5                [K_VS_T, K_VS_VT]:
```

```
;  5244          5347  6                 BEGIN
;  5245          5348  6                 .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5246          5349  6                 IF .TORN EQL 0
;  5247          5350  6                 THEN
;  5248          5351  7                     BEGIN
;  5249          5352  7                     SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER] + 2;
;  5250          5353  7                     SRC_INFO [S_LEN] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 16, 0;, BYTE];
;  5251          5354  7                     END
;  5252          5355  6                 ELSE
;  5253          5356  6                     DST_INFO [D_LEN] = .DESTINATION [DSC$W_LENGTH];
;  5254          5357  5                 END;
;  5255          5358  5
;  5256          5359  5             [K_VS_AC]:
;  5257          5360  6                 BEGIN
;  5258          5361  6                 .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5259          5362  6                 IF .TORN EQL 0
;  5260          5363  6                 THEN
;  5261          5364  7                     BEGIN
;  5262          5365  7                     SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER] + 1;
;  5263          5366  7                     SRC_INFO [S_LEN] = .BLOCK [.SOURCE [DSC$A_POINTER], 0, 0, 8, 0;, BYTE];
;  5264          5367  7                     END
;  5265          5368  6                 ELSE
;  5266          5369  6                     DST_INFO [D_LEN] = .DESTINATION [DSC$W_LENGTH];
;  5267          5370  5                 END;
;  5268          5371  5
;  5269          5372  5             [K_VS_AZ]:
;  5270          5373  6                 BEGIN
;  5271          5374  6                 .LEFT_OR_RIGHT_CVT = K_NBDS;
;  5272          5375  6                 IF .TORN EQL 0
;  5273          5376  6                 THEN
;  5274          5377  7                     BEGIN
;  5275          5378  7                     LOCAL
;  5276          5379  7                         SRC_PTR:  REF VECTOR[, BYTE],
;  5277          5380  7                         COUNT;
;  5278          5381  7                     COUNT = 0;
;  5279          5382  7                     SRC_PTR = .SOURCE[DSC$A_POINTER];
;  5280          5383  7                     WHILE .SRC_PTR[.COUNT] NEQ 0 DO
;  5281          5384  7                         COUNT = .COUNT + 1;
;  5282          5385  7                     SRC_INFO[S_LEN] = .COUNT;
;  5283          5386  7                     SRC_INFO [S_POINTER] = .SOURCE [DSC$A_POINTER];
;  5284          5387  7                     END
;  5285          5388  6                 ELSE
;  5286          5389  6                     DST_INFO [D_LEN] = .DESTINATION [DSC$W_LENGTH];
;  5287          5390  5                 END;
;  5288          5391  5
;  5289          5392  5             [INRANGE, OUTRANGE]:
;  5290          5393  5                 $DBG_ERROR ('DBGCVTDX\FIND_CVT_PATH:  invalid final state');
;  5291          5394  5             TES;
;  5292          5395  5         END
;  5293          5396  4     )                                       ! End of INCRU, with a EXITLOOP value.
;  5294          5397  2     END;                                    ! End of STATUS.
;  5295          5398  2
;  5296          5399  2
;  5297          5400  2     ! Map the left and right of the conversion, (i.e. if the conversion is
;  5298          5401  2     ! K_SMLINT_LRGFLTCMPLX, then LEFT_CVT is SMLINT and RIGHT_CVT is LRGFLTCMPLX)
;  5299          5402  2     ! into a final conversion index and return with the status of this routine.
;  5300          5403  2     !
```

```
;  5301            5404  2      .CVT_PATH = (.LEFT_CVT - 1)*K_TOT_CAT + .RIGHT_CVT;
;  5302            5405  2      RETURN .STATUS;
;  5303            5406  1      END;                                        ! End of routine FIND_CVT_PATH
;  INFO#250                    L1:5404
;  Referenced LOCAL symbol LEFT_CVT is probably not initialized
;  INFO#250                    L1:5404
;  Referenced LOCAL symbol RIGHT_CVT is probably not initialized


                                                                .PSECT  DBG$PLIT,NOWRT,  SHR,  PIC,0

76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  019B0 P.AKM:  .ASCII  <24>\DBGCVTDX:  invalid class\                ;
                    73  73  61  6C  63  20  64  69  6C  61  019BF
76  6E  69  20  20  3A  58  44  54  56  43  47  42  44  18  019C9 P.AKN:  .ASCII  <24>\DBGCVTDX:  invalid class\                ;
                    73  73  61  6C  63  20  64  69  6C  61  019D8
5F  44  4E  49  46  5C  58  44  54  56  43  47  42  44  26  019E2 P.AKO:  .ASCII  \&DBGCVTDX\<92>\FIND_CVT_PATH:  invalid \   ;
61  76  6E  69  20  20  3A  48  54  41  50  5F  54  56  43  019F1
                                            20  64  69  6C  01A00
                                        65  74  61  74  73  01A04          .ASCII  \state\                                     ;
                                                            01A09          .BLKB   3
                                            00  00  00  0C  01A0C P.AKP:  .ASCII  <12><0><0><0>                              ;
5F  44  4E  49  46  5C  58  44  54  56  43  47  42  44  16  01A10 P.AKQ:  .ASCII  <22>\DBGCVTDX\<92>\FIND_CVT_PATH\           ;
                        48  54  41  50  5F  54  56  43  01A1F
5F  44  4E  49  46  5C  58  44  54  56  43  47  42  44  2C  01A27 P.AKR:  .ASCII  \,DBGCVTDX\<92>\FIND_CVT_PATH:  invalid \   ;
61  76  6E  69  20  20  3A  48  54  41  50  5F  54  56  43  01A36
                                            20  64  69  6C  01A45
                65  74  61  74  73  20  6C  61  6E  69  66  01A49          .ASCII  \final state\                               ;


                                                                .PSECT  DBG$CODE,NOWRT,  SHR,  PIC,0

                                            0FFC 00000 FIND_CVT_PATH:
                                                                .WORD   Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11                ; 4435
                    5E          C0  AE  9E 00002                MOVAB   -64(SP), SP                                         ;
                                0C  AE  D4 00006                CLRL    TURN                                                ; 4597
                02          00  0C  AE  CF 00009 1$:            CASEL   TURN, #0, #2                                        ; 4605
              003C        0021      0006      0000E 2$:         .WORD   3$-2$,-                                             ;
                                                                        4$-2$,-                                             ;
                                                                        5$-2$                                               ;
                    50          04  AC  D0 00014 3$:            MOVL    SOURCE, R0                                          ; 4609
                04  AE          03  A0  9A 00018                MOVZBL  3(R0), CLASS                                        ;
                    6E          02  A0  9A 0001D                MOVZBL  2(R0), DTYPE                                        ; 4610
                    59              50  D0 00021                MOVL    R0, SRC_OR_DST                                      ; 4611
                    5A          0C  AC  D0 00024                MOVL    SRC_INFO, SRC_OR_DST_INFO                           ; 4612
                08  AE          18  AE  9E 00028                MOVAB   LEFT_CVT, LEFT_OR_RIGHT_CVT                         ; 4613
                                    20  11 0002D                BRB     6$                                                  ; 4605
                    50          08  AC  D0 0002F 4$:            MOVL    DESTINATION, R0                                     ; 4617
                04  AE          03  A0  9A 00033                MOVZBL  3(R0), CLASS                                        ;
                    6E          02  A0  9A 00038                MOVZBL  2(R0), DTYPE                                        ; 4618
                    59              50  D0 0003C                MOVL    R0, SRC_OR_DST                                      ; 4619
                    5A          10  AC  D0 0003F                MOVL    DST_INFO, SRC_OR_DST_INFO                           ; 4620
                08  AE          1C  AE  9E 00043                MOVAB   RIGHT_CVT, LEFT_OR_RIGHT_CVT                        ; 4621
                                    05  11 00048                BRB     6$                                                  ; 4605
                    50              01  D0 0004A 5$:            MOVL    #1, STATUS                                          ; 4624
                                    6E  11 0004D                BRB     12$                                                 ;
```

```
                                    0E        04    AE  D1 0004F 6$:     CMPL    CLASS, #14                                  ; 4630
                                                    05  1B 00053         BLEQU   7$                                          ;
                                    50              05  CE 00055         MNEGL   #5, STATUS                                  ;
                                                    63  11 00058         BRB     12$                                         ;
                                    2A              6E  D1 0005A 7$:      CMPL    DTYPE, #42                                  ; 4631
                                                    05  1B 0005D         BLEQU   8$                                          ;
                                    50              06  CE 0005F         MNEGL   #6, STATUS                                  ;
                                                    59  11 00062         BRB     12$                                         ;
                                    50        04    AE  D0 00064 8$:      MOVL    CLASS, R0                                   ; 4636
                                    56  00000000'EF40  98 00068         CVTBL   CLASS_TABLE[R0], STATE                      ;
                              14 FFFFFFF9       8F    56  CF 00070         CASEL   STATE, #-7, #20                             ; 4641
                0042          0042          0042        0042  00078 9$:      .WORD   11$-9$,-                                    ;
                002A          0042          0042        0042  00080                 11$-9$,-                                    ;
                0047          0047          0047        0047  00088                 11$-9$,-                                    ;
                0047          0047          0047        0047  00090                 11$-9$,-                                    ;
                0047          0047          0047        0047  00098                 11$-9$,-                                    ;
                                                        0047  000A0                 11$-9$,-                                    ;
                                                                                    10$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$,-                                    ;
                                                                                    13$-9$                                      ;
                                    00000000'     EF  9F 000A2 10$:     PUSHAB  P.AKO                                       ; 4671
                                                  01  DD 000A8         PUSHL   #1                                          ;
                                    00028362      8F  DD 000AA         PUSHL   #164706                                     ;
                          00000000G   00          03  FB 000B0         CALLS   #3, LIB$SIGNAL                              ;
                                                00CC  31 000B7         BRW     36$                                         ;
                                    50            56  D0 000BA 11$:     MOVL    STATE, STATUS                               ; 4649
                                                  68  11 000BD 12$:     BRB     24$                                         ;
                          0C            01        56  CF 000BF 13$:     CASEL   STATE, #1, #12                              ; 4658
                0038          001A          0033        002F  000C3 14$:     .WORD   16$-14$,-                                   ;
                001A          001A          001A        001A  000CB                 17$-14$,-                                   ;
                001A          0047          0042        003D  000D3                 15$-14$,-                                   ;
                                                        004C  000DB                 18$-14$,-                                   ;
                                                                                    15$-14$,-                                   ;
                                                                                    15$-14$,-                                   ;
                                                                                    15$-14$,-                                   ;
                                                                                    15$-14$,-                                   ;
                                                                                    19$-14$,-                                   ;
                                                                                    20$-14$,-                                   ;
                                                                                    21$-14$,-                                   ;
                                                                                    15$-14$,-                                   ;
                                                                                    22$-14$                                     ;
                                    00000000'     EF  9F 000DD 15$:     PUSHAB  P.AKM                                       ;
                                                  01  DD 000E3         PUSHL   #1                                          ;
                                    00028362      8F  DD 000E5         PUSHL   #164706                                     ;
                          00000000G   00          03  FB 000EB         CALLS   #3, LIB$SIGNAL                              ;
```

```
                                                50  D4 000F2 16$:   CLRL    R0                                      ;
                                                1C  11 000F4        BRB     23$                                     ;
                                      50        01  D0 000F6 17$:   MOVL    #1, R0                                  ;
                                                17  11 000F9        BRB     23$                                     ;
                                      50        02  D0 000FB 18$:   MOVL    #2, R0                                  ;
                                                12  11 000FE        BRB     23$                                     ;
                                      50        03  D0 00100 19$:   MOVL    #3, R0                                  ;
                                                0D  11 00103        BRB     23$                                     ;
                                      50        04  D0 00105 20$:   MOVL    #4, R0                                  ;
                                                08  11 00108        BRB     23$                                     ;
                                      50        05  D0 0010A 21$:   MOVL    #5, R0                                  ;
                                                03  11 0010D        BRB     23$                                     ;
                                      50        06  D0 0010F 22$:   MOVL    #6, R0                                  ;
                                      50        2B  C4 00112 23$:   MULL2   #43, R0                                 ;
                                      50 00000000'EF40  9E 00115    MOVAB   DTYPE_TABLE[R0], R0                     ;
                                      57     00 BE40  98 00118    CVTBL   @DTYPE[R0], TOKEN                       ;
                                                06  18 00122        BGEQ    25$                                     ; 4663
                                      50        57  D0 00124        MOVL    TOKEN, STATUS                           ;
                                              0A03  31 00127 24$:   BRW     167$                                    ;
                                   0C         01  56  CF 0012A 25$:   CASEL   STATE, #1, #12                         ; 4668
                0021               003A       001C    004F  0012E 26$:   .WORD  34$-26$,-                           ;
                003A               003A       003A    003A  00136        27$-26$,-                                   ;
                003A               0030       002B    0026  0013E        33$-26$,-                                   ;
                                              0035    0035  00146        28$-26$,-                                   ;
                                                                          33$-26$,-                                   ;
                                                                          33$-26$,-                                   ;
                                                                          33$-26$,-                                   ;
                                                                          33$-26$,-                                   ;
                                                                          29$-26$,-                                   ;
                                                                          30$-26$,-                                   ;
                                                                          31$-26$,-                                   ;
                                                                          33$-26$,-                                   ;
                                                                          32$-26$                                     ;
                                                33  11 00148        BRB     34$                                     ;
                                      50        01  D0 0014A 27$:   MOVL    #1, R0                                  ;
                                                30  11 0014D        BRB     35$                                     ;
                                      50        02  D0 0014F 28$:   MOVL    #2, R0                                  ;
                                                2B  11 00152        BRB     35$                                     ;
                                      50        03  D0 00154 29$:   MOVL    #3, R0                                  ;
                                                26  11 00157        BRB     35$                                     ;
                                      50        04  D0 00159 30$:   MOVL    #4, R0                                  ;
                                                21  11 0015C        BRB     35$                                     ;
                                      50        05  D0 0015E 31$:   MOVL    #5, R0                                  ;
                                                1C  11 00161        BRB     35$                                     ;
                                      50        06  D0 00163 32$:   MOVL    #6, R0                                  ;
                                                17  11 00166        BRB     35$                                     ;
                                         00000000'  EF  9F 00168 33$:   PUSHAB  P.AKN                               ;
                                                01  DD 0016E        PUSHL   #1                                      ;
                                         00028362   8F  DD 00170        PUSHL   #164706                             ;
                                00000000G     00        03  FB 00176        CALLS   #3, LIB$SIGNAL                  ;
                                                50  D4 0017D 34$:   CLRL    R0                                      ;
                                      50        2B  C4 0017F 35$:   MULL2   #43, R0                                 ;
                                   56           50  57  C1 00182        ADDL3   TOKEN, R0, STATE                    ;
                         0000012B  8F           01  56  CF 00186 36$:   CASEL   STATE, #1, #299                     ; 4690
                02D1               02A0       026F    0353  0018E 37$:   .WORD  56$-37$,-                           ;
                0346               0315       02E4    039E  00196        39$-37$,-                                   ;
                04A0               04DC       046F    039E  0019E        41$-37$,-                                   ;
```

```
                0603    05C6    05A2    04EB    001A6                   44$-37$,-
                0835    07F8    07A7    0649    001AE                   64$-37$,-
                0258    0258    08B0    0872    001B6                   47$-37$,-
                053E    04FA    0403    0258    001BE                   51$-37$,-
                0258    0258    0570    0531    001C6                   55$-37$,-
                0258    0258    0258    0258    001CE                   64$-37$,-
                0353    0258    0258    0258    001D6                   80$-37$,-
                0258    0258    0258    0353    001DE                   87$-37$,-
                0258    0258    0258    025B    001E6                   84$-37$,-
                0258    0258    0258    0258    001EE                   88$-37$,-
                0258    0258    0258    0258    001F6                   102$-37$,-
                0258    0258    0258    08B6    001FE                   104$-37$,-
                0258    0258    0258    025B    00206                   107$-37$,-
                0258    0258    0258    0258    0020E                   113$-37$,-
                0258    0258    0258    025B    00216                   135$-37$,-
                0258    0258    0258    0258    0021E                   140$-37$,-
                0258    0258    0258    0258    00226                   143$-37$,-
                0258    0258    0258    025B    0022E                   146$-37$,-
                08C0    0258    0258    0258    00236                   149$-37$,-
                0258    0258    0258    0258    0023E                   38$-37$,-
                0258    0258    0258    0258    00246                   38$-37$,-
                08C0    0258    0258    0258    0024E                   38$-37$,-
                0258    0258    0258    0258    00256                   71$-37$,-
                0258    0258    0258    0258    0025E                   90$-37$,-
                0258    0258    0258    0258    00266                   95$-37$,-
                0258    0258    0258    0258    0026E                   94$-37$,-
                0258    0258    0258    0258    00276                   98$-37$,-
                0258    0258    0258    025B    0027E                   38$-37$,-
                0258    0258    0258    025B    00286                   38$-37$,-
                02A0    026F    0258    025B    0028E                   38$-37$,-
                0319    02E4    039E    02D1    00296                   38$-37$,-
                04DC    046F    039E    0346    0029E                   38$-37$,-
                05C6    05A2    04EB    04A0    002A6                   38$-37$,-
                07F8    07A7    0649    0603    002AE                   38$-37$,-
                0258    0258    0872    0835    002B6                   38$-37$,-
                04FA    0403    0258    0258    002BE                   38$-37$,-
                0258    0570    0531    053E    002C6                   56$-37$,-
                0258    0258    0258    0258    002CE                   56$-37$,-
                0258    0258    0258    0258    002D6                   38$-37$,-
                0258    0258    0258    0258    002DE                   38$-37$,-
                0258    0258    08C0    0258    002E6                   38$-37$,-
                0258    0258    0258    0258    002EE                   38$-37$,-
                0258    0258    0258    0258    002F6                   38$-37$,-
                0258    0258    08C0    0258    002FE                   38$-37$,-
                0258    0258    0258    0258    00306                   38$-37$,-
                0258    0258    0258    0258    0030E                   38$-37$,-
                0258    0258    0258    0258    00316                   38$-37$,-
                0258    0258    0258    0258    0031E                   38$-37$,-
                0258    0258    0258    0258    00326                   38$-37$,-
                0258    0258    0258    0258    0032E                   38$-37$,-
                0258    0258    0258    0258    00336                   38$-37$,-
                0258    0258    0258    0258    0033E                   38$-37$,-
                0258    0258    0258    0258    00346                   38$-37$,-
                0258    0258    0258    0258    0034E                   151$-37$,-
                0258    0258    0258    0920    00356                   38$-37$,-
                0258    0258    0258    0258    0035E                   38$-37$,-
                0258    0258    0258    0258    00366                   38$-37$,-
```

```
        0258        025B        025B        0258    0036E            38$-37$,-
        0258        025B        025B        0258    00376            38$-37$,-
        0920        025B        025B        0258    0037E            38$-37$,-
        0258        025B        095C        093E    00386            38$-37$,-
        026F        025B        025B        0258    0038E            38$-37$,-
        02E4        039E        02D1        02A0    00396            38$-37$,-
        046F        039E        0346        0315    0039E            38$-37$,-
        05A2        04EB        04A0        04DC    003A6            38$-37$,-
        0258        025B        025B        0258    003AE            38$-37$,-
        0258        025B        025B        0258    003B6            38$-37$,-
        0403        025B        025B        0258    003BE            38$-37$,-
        0370        0531        053E        04FA    003C6            38$-37$,-
        0353        025B        025B        0258    003CE            38$-37$,-
        0258        025B        025B        0258    003D6            38$-37$,-
        0353        025B        0353        0258    003DE            38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     152$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     152$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
                                                                     38$-37$,-
```

```
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
36$-37$,-
39$-37$,-
41$-37$,-
44$-37$,-
64$-37$,-
47$-37$,-
51$-37$,-
55$-37$,-
64$-37$,-
80$-37$,-
87$-37$,-
84$-37$,-
88$-37$,-
102$-37$,-
104$-37$,-
107$-37$,-
113$-37$,-
135$-37$,-
140$-37$,-
143$-37$,-
146$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
71$-37$,-
90$-37$,-
95$-37$,-
94$-37$,-
98$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
152$-37$,-
```

```
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
152$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
38$-37$,-
159$-37$,-
38$-37$,-
38$-37$,-
```

```
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                159$-37$,-
                160$-37$,-
                161$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                39$-37$,-
                41$-37$,-
                44$-37$,-
                64$-37$,-
                47$-37$,-
                51$-37$,-
                55$-37$,-
                64$-37$,-
                80$-37$,-
                87$-37$,-
                84$-37$,-
                88$-37$,-
                102$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                38$-37$,-
                71$-37$,-
                90$-37$,-
                95$-37$,-
                94$-37$,-
                98$-37$,-
```

```
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        56$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        56$-37$,-                         ;
                                                                        38$-37$,-                         ;
                                                                        56$-37$                           ;
                                      00000000'   EF    9F 003E6 38$:   PUSHAB  P.AKR                     ; 5393
                                                  01    DD 003EC        PUSHL   #1                        ;
                                      00028362    8F    DD 003EE        PUSHL   #164706                   ;
                        00000000G  00             03    FB 003F4        CALLS   #3, LIB$SIGNAL            ;
                                                  60    11 003FB        BRB     43$                       ;
                               08  BE             01    D0 003FD 39$:   MOVL    #1, @LEFT_OR_RIGHT_CVT    ; 4695
                         00000083  8F             56    D1 00401        CMPL    STATE, #131               ; 4696
                                                  10    12 00408        BNEQ    40$                       ;
                                   6A       08    A9    90 0040A        MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)  ; 4699
          50         0A    A9                     01    03    EF 0040E  EXTZV   #3, #1, 10(SRC_OR_DST), R0 ; 4701
   07     AA               01                     01    50    F0 00414  INSV    R0, #1, #1, 7(SRC_OR_DST_INFO) ;
                                            0C    AE    D5 0041A 40$:   TSTL    TURN                      ; 4703
                                                  73    12 0041D        BNEQ    49$                       ;
                                   51       0C    AC    D0 0041F        MOVL    SRC_INFO, R1              ; 4705
                                   50       04    AC    D0 00423        MOVL    SOURCE, R0                ;
                               01  B1       04    B0    9A 00427        MOVZBL  @4(R0), @1(R1)            ;
                                                  73    11 0042C        BRB     50$                       ; 4690
                               08  BE             01    D0 0042E 41$:   MOVL    #1, @LEFT_OR_RIGHT_CVT    ; 4711
                         00000084  8F             56    D1 00432        CMPL    STATE, #132               ; 4712
                                                  10    12 00439        BNEQ    42$                       ;
                                   6A       08    A9    90 0043B        MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)  ; 4715
          50         0A    A9                     01    03    EF 0043F  EXTZV   #3, #1, 10(SRC_OR_DST), R0 ; 4717
   07     AA               01                     01    50    F0 00445  INSV    R0, #1, #1, 7(SRC_OR_DST_INFO) ;
                                            0C    AE    D5 0044B 42$:   TSTL    TURN                      ; 4719
                                                  73    12 0044E        BNEQ    53$                       ;
                                   51       0C    AC    D0 00450        MOVL    SRC_INFO, R1              ; 4721
                                   50       04    AC    D0 00454        MOVL    SOURCE, R0                ;
                               01  B1       04    B0    3C 00458        MOVZWL  @4(R0), @1(R1)            ;
                                                  73    11 0045D 43$:   BRB     54$                       ; 4690
                               08  BE             02    D0 0045F 44$:   MOVL    #2, @LEFT_OR_RIGHT_CVT    ; 4727
                         00000085  8F             56    D1 00463        CMPL    STATE, #133               ; 4728
                                                  03    13 0046A 45$:   BEQL    46$                       ;
                                                01AB    31 0046C        BRW     82$                       ;
                                                0198    31 0046F 46$:   BRW     81$                       ;
                               08  BE             01    D0 00472 47$:   MOVL    #1, @LEFT_OR_RIGHT_CVT    ; 4743
                         00000087  8F             56    D1 00476        CMPL    STATE, #135               ; 4744
                                                  10    12 0047D        BNEQ    48$                       ;
                                   6A       08    A9    90 0047F        MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)  ; 4747
          50         0A    A9                     01    03    EF 00483  EXTZV   #3, #1, 10(SRC_OR_DST), R0 ; 4749
   07     AA               01                     01    50    F0 00489  INSV    R0, #1, #1, 7(SRC_OR_DST_INFO) ;
                                            0C    AE    D5 0048F 48$:   TSTL    TURN                      ; 4751
                                                  58    12 00492 49$:   BNEQ    57$                       ;
                                   51       0C    AC    D0 00494        MOVL    SRC_INFO, R1              ; 4753
                                   50       04    AC    D0 00498        MOVL    SOURCE, R0                ;
                               01  B1       04    B0    98 0049C        CVTBL   @4(R0), @1(R1)            ;
```

```
                                                79  11  004A1  50$:    BRB     61$                                     ;4690
                                    08  BE      01  D0  004A3  51$:    MOVL    #1, @LEFT_OR_RIGHT_CVT                  ;4759
                          00000088  8F          56  D1  004A7          CMPL    STATE, #136                             ;4760
                                                10  12  004AE          BNEQ    52$                                     ;
                                    6A      08  A9  90  004B0          MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ;4763
                 50         0A      A9          01  03  EF  004B4          EXTZV   #3, #1, 10(SRC_OR_DST), R0              ;4765
          07     AA         01      01          50  F0  004BA          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)          ;
                                                0C  AE  D5  004C0  52$:    TSTL    TURN                                    ;4767
                                                65  12  004C3  53$:    BNEQ    63$                                     ;
                                    51  0C      AC  D0  004C5          MOVL    SRC_INFO, R1                            ;4769
                                    50  04      AC  D0  004C9          MOVL    SOURCE, R0                              ;
                                01  B1  04      B0  32  004CD          CVTWL   @4(R0), @1(R1)                          ;
                                                56  11  004D2  54$:    BRB     63$                                     ;4690
                                    08  BE      01  D0  004D4  55$:    MOVL    #1, @LEFT_OR_RIGHT_CVT                  ;4775
                          00000089  8F          56  D1  004D8          CMPL    STATE, #137                             ;4776
                                                89  11  004DF          BRB     45$                                     ;
                                    08  BE      01  D0  004E1  56$:    MOVL    #1, @LEFT_OR_RIGHT_CVT                  ;4791
                                    05  AA      69  B0  004E5          MOVW    (SRC_OR_DST), 5(SRC_OR_DST_INFO)        ;4792
                                                0C  AE  D5  004E9          TSTL    TURN                                    ;4793
                                                67  12  004EC  57$:    BNEQ    67$                                     ;
                                    51  04      AC  D0  004EE          MOVL    SOURCE, R1                              ;4799
                                    0D  03      A1  91  004F2          CMPB    3(R1), #13                              ;
                                                06  12  004F6          BNEQ    58$                                     ;
                                    52  08      A1  D0  004F8          MOVL    8(R1), BITPOS                           ;4801
                                                02  11  004FC          BRB     59$                                     ;
                                                52  D4  004FE  58$:    CLRL    BITPOS                                  ;4803
                                    53  04      A1  D0  00500  59$:    MOVL    4(R1), SRC_PTR                          ;4804
                                    50  0C      AC  D0  00504          MOVL    SRC_INFO, R0                            ;4807
                                    29          56  D1  00508          CMPL    STATE, #41                              ;4805
                                                09  13  0050B          BEQL    60$                                     ;
                          0000012C  8F          56  D1  0050D          CMPL    STATE, #300                             ;
                                                08  12  00514          BNEQ    62$                                     ;
          01     B0         63      61          52  EE  00516  60$:    EXTV    BITPOS, (R1), (SRC_PTR), @1(R0)         ;4807
                                                71  11  0051C  61$:    BRB     70$                                     ;
          01     B0         63      61          52  EF  0051E  62$:    EXTZV   BITPOS, (R1), (SRC_PTR), @1(R0)         ;4810
                                                69  18  00524          BGEQ    70$                                     ;4811
                                    08  BE      02  D0  00526          MOVL    #2, @LEFT_OR_RIGHT_CVT                  ;4813
                                                63  11  0052A  63$:    BRB     70$                                     ;4793
                                    08  BE      02  D0  0052C  64$:    MOVL    #2, @LEFT_OR_RIGHT_CVT                  ;4820
                          0000008A  8F          56  D1  00530          CMPL    STATE, #138                             ;4821
                                                09  13  00537          BEQL    65$                                     ;
                          00000086  8F          56  D1  00539          CMPL    STATE, #134                             ;
                                                10  12  00540          BNEQ    66$                                     ;
                                    6A      08  A9  90  00542  65$:    MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ;4825
                 50         0A      A9          01  03  EF  00546          EXTZV   #3, #1, 10(SRC_OR_DST), R0              ;4827
          07     AA         01      01          50  F0  0054C          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)          ;
                                                0C  AE  D5  00552  66$:    TSTL    TURN                                    ;4829
                                                5A  12  00555  67$:    BNEQ    73$                                     ;
                                    53  0C      AC  D0  00557          MOVL    SRC_INFO, R3                            ;4832
                                    50  01      A3  D0  0055B          MOVL    1(R3), R0                               ;
                                    51  04      AC  D0  0055F          MOVL    SOURCE, R1                              ;
                                    52  04      A1  D0  00563          MOVL    4(R1), R2                               ;
                                    60          62  D0  00567          MOVL    (R2), (R0)                              ;
                                    51  04      A0  9E  0056A          MOVAB   4(R0), R1                               ;4833
                                    61  04      A2  D0  0056E          MOVL    4(R2), (R1)                             ;
                                                53  18  00572          BGEQ    74$                                     ;4834
                                    60          60  D2  00574          MCOML   (R0), (R0)                              ;4837
```

```
                                                61    61  D2 00577          MCOML   (R1), (R1)                                      ; 4838
                                FFFFFFFF        8F    60  D1 0057A          CMPL    (R0), #-1                                       ; 4839
                                                      06  12 00581          BNEQ    68$                                             ;
                                                      60  D4 00583          CLRL    (R0)                                            ; 4842
                                                      61  D6 00585          INCL    (R1)                                            ; 4843
                                                      02  11 00587          BRB     69$                                             ; 4839
                                                      60  D6 00589 68$:      INCL    (R0)                                            ; 4846
                                        07      A3    01  88 0058B 69$:      BISB2   #1, 7(R3)                                       ; 4847
                                                      6A  11 0058F 70$:      BRB     79$                                             ; 4690
                                        08      BE    02  D0 00591 71$:      MOVL    #2, @LEFT_OR_RIGHT_CVT                          ; 4854
                                0000009B        8F    56  D1 00595          CMPL    STATE, #155                                     ; 4855
                                                      10  12 0059C          BNEQ    72$                                             ;
                                                6A    08  A9 90 0059E          MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 4858
        50      0A      A9                      01  03  EF 005A2          EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 4860
07      AA              01                      01  50  F0 005A8          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                                0C  AE  D5 005AE 72$:      TSTL    TURN                                            ; 4862
                                                      79  12 005B1 73$:      BNEQ    83$                                             ;
                                                50    04  AC  D0 005B3          MOVL    SOURCE, R0                                      ; 4865
                                                5B    0C  AC  D0 005B7          MOVL    SRC_INFO, R11                                   ;
                                                58    01  AB  D0 005BB          MOVL    1(R11), R8                                      ;
                                        68      D4  B0    10  28 005BF          MOVC3   #16, @4(R0), (R8)                               ;
                                                0F      A8  95 005C4          TSTB    15(R8)                                          ; 4866
                                                      63  18 005C7 74$:      BGEQ    83$                                             ;
                                                      50  D4 005C9          CLRL    I                                               ; 4869
                                                    6048  9F 005CB 75$:      PUSHAB  (I)[R8]                                         ; 4870
                                                    6048  9F 005CE          PUSHAB  (I)[R8]                                         ;
                                                9E      9E  D2 005D1          MCOML   @(SP)+, @(SP)+                                  ;
        FFF1            50                      04  0C  F1 005D4          ACBL    #12, #4, I, 75$                                 ;
                                FFFFFFFF        8F    68  D1 005DA          CMPL    (R8), #-1                                       ; 4871
                                                      12  12 005E1          BNEQ    77$                                             ;
                                                      68  D4 005E3          CLRL    (R8)                                            ; 4874
                                                50    04  D0 005E5          MOVL    #4, I                                           ; 4875
                                                    6048  9F 005E8 76$:      PUSHAB  (I)[R8]                                         ; 4876
                                                9E      9E  D6 005EB          INCL    @(SP)+                                          ;
        FFF5            50                      04  0C  F1 005ED          ACBL    #12, #4, I, 76$                                 ;
                                                      02  11 005F3          BRB     78$                                             ; 4871
                                                      68  D6 005F5 77$:      INCL    (R8)                                            ; 4879
                                        07      AB    01  88 005F7 78$:      BISB2   #1, 7(R11)                                      ; 4880
                                                      6B  11 005FB 79$:      BRB     86$                                             ; 4690
                                        08      BE    03  D0 005FD 80$:      MOVL    #3, @LEFT_OR_RIGHT_CVT                          ; 4887
                                0000008B        8F    56  D1 00601          CMPL    STATE, #139                                     ; 4888
                                                      10  12 00608          BNEQ    82$                                             ;
                                                6A    08  A9 90 0060A 81$:      MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 4891
        50      0A      A9                      01  03  EF 0060E          EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 4893
07      AA              01                      01  50  F0 00614          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                                0C  AE  D5 0061A 82$:      TSTL    TURN                                            ; 4895
                                                      49  12 0061D          BNEQ    86$                                             ;
                                                51    0C  AC  D0 0061F          MOVL    SRC_INFO, R1                                    ; 4897
                                                50    04  AC  D0 00623          MOVL    SOURCE, R0                                      ;
                                        01      B1    04  B0  D0 00627          MOVL    @4(R0), @1(R1)                                  ;
                                                      3A  11 0062C 83$:      BRB     86$                                             ; 4690
                                        08      BE    03  D0 0062E 84$:      MOVL    #3, @LEFT_OR_RIGHT_CVT                          ; 4902
                                0000008D        8F    56  D1 00632          CMPL    STATE, #141                                     ; 4903
                                                      10  12 00639          BNEQ    85$                                             ;
                                                6A    08  A9 90 0063B          MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 4906
        50      0A      A9                      01  03  EF 0063F          EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 4908
07      AA              01                      01  50  F0 00645          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
```

```
                                                0C   AE   D5 0064B 85$:    TSTL    TURN                                    ; 4910
                                                     6D   12 0064E         BNEQ    93$
                                          50    0C   AC   D0 00650         MOVL    SRC_INFO, R0                            ; 4913
                                          51    01   A0   D0 00654         MOVL    1(R0), R1
                                          50    04   AC   D0 00658         MOVL    SOURCE, R0
                                          50    04   A0   D0 0065C         MOVL    4(R0), R0
                                          61    00   60   D0 00660         MOVL    (R0), (R1)
                                    08    A1    04   A0   D0 00663         MOVL    4(R0), 8(R1)                            ; 4918
                                                     53   11 00668 86$:    BRB     93$                                     ; 4690
                                    08    BE          03   D0 0066A 87$:    MOVL    #3, @LEFT_OR_RIGHT_CVT                  ; 4924
                              0000008C    8F          56   D1 0066E         CMPL    STATE, #140                             ; 4925
                                                     1E   13 00675         BEQL    91$
                                                     2C   11 00677         BRB     92$                                     ; 4932
                                    08    BE          03   D0 00679 88$:    MOVL    #3, @LEFT_OR_RIGHT_CVT                  ; 4948
                              0000008C    8F          56   D1 0067D         CMPL    STATE, #140                             ; 4949
                                                     53   13 00684 89$:    BEQL    96$
                                                     61   11 00686         BRB     97$                                     ; 4956
                                    08    BE          04   D0 00688 90$:    MOVL    #4, @LEFT_OR_RIGHT_CVT                  ; 4963
                              0000009C    8F          56   D1 0068C         CMPL    STATE, #156                             ; 4964
                                                     10   12 00693         BNEQ    92$
                                          6A    08    A9   90 00695 91$:    MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 4967
          50        0A    A9                      01   03   EF 00699         EXTZV   #3, #1, 10(SRC_OR_DST), R0              ; 4969
    07    AA              01                      01   50   F0 0069F         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                0C   AE   D5 006A5 92$:    TSTL    TURN                                    ; 4971
                                                     74   12 006A8         BNEQ    100$
                                          50    0C   AC   D0 006AA         MOVL    SRC_INFO, R0                            ; 4974
                                          51    01   A0   D0 006AE         MOVL    1(R0), R1
                                          50    04   AC   D0 006B2         MOVL    SOURCE, R0
                                          50    04   A0   D0 006B6         MOVL    4(R0), R0
                                          61          60   7D 006BA         MOVQ    (R0), (R1)
                                                     6F   11 006BD 93$:    BRB     101$                                    ; 4690
                                    08    BE          04   D0 006BF 94$:    MOVL    #4, @LEFT_OR_RIGHT_CVT                  ; 4981
                              0000009E    8F          56   D1 006C3         CMPL    STATE, #158                             ; 4982
                                                     B8   11 006CA         BRB     89$
                                    08    BE          04   D0 006CC 95$:    MOVL    #4, @LEFT_OR_RIGHT_CVT                  ; 4996
                              0000009D    8F          56   D1 006D0         CMPL    STATE, #157                             ; 4997
                                                     10   12 006D7         BNEQ    97$
                                          6A    08    A9   90 006D9 96$:    MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 5000
          50        0A    A9                      01   03   EF 006DD         EXTZV   #3, #1, 10(SRC_OR_DST), R0              ; 5002
    07    AA              01                      01   50   F0 006E3         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                0C   AE   D5 006E9 97$:    TSTL    TURN                                    ; 5004
                                                     40   12 006EC         BNEQ    101$
                                          51    04   AC   D0 006EE         MOVL    SOURCE, R1
                                          50    0C   AC   D0 006F2         MOVL    SRC_INFO, R0
                                    01    B0    04   B1   10 28 006F6         MOVC3   #16, @4(R1), @1(R0)
                                                     30   11 006FC         BRB     101$                                    ; 4690
                                    08    BE          04   D0 006FE 98$:    MOVL    #4, @LEFT_OR_RIGHT_CVT                  ; 5009
                              0000009F    8F          56   D1 00702         CMPL    STATE, #159                             ; 5010
                                                     10   12 00709         BNEQ    99$
                                          6A    08    A9   90 0070B         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 5013
          50        0A    A9                      01   03   EF 0070F         EXTZV   #3, #1, 10(SRC_OR_DST), R0              ; 5015
    07    AA              01                      01   50   F0 00715         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                0C   AE   D5 0071B 99$:    TSTL    TURN                                    ; 5017
                                                     6F   12 0071E 100$:    BNEQ    106$
                                          51    04   AC   D0 00720         MOVL    SOURCE, R1                              ; 5019
                                          50    0C   AC   D0 00724         MOVL    SRC_INFO, R0
                                    01    B0    04   B1   20 28 00728         MOVC3   #32, @4(R1), @1(R0)
```

```
                                        5F  11 0072E 101$:   BRB     106$                                            ; 4690
                        08   BE         06  D0 00730 102$:   MOVL    #6, @LEFT_OR_RIGHT_CVT                          ; 5024
                        05   AA         69  B0 00734         MOVW    (SRC_OR_DST), 5(SRC_OR_DST_INFO)                ; 5025
                  0000008F   8F         56  D1 00738         CMPL    STATE, #143                                     ; 5026
                             10         12  0073F            BNEQ    103$                                            ;
                             6A    08   A9  90 00741         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 5029
      50      0A   A9        01         03  EF 00745         EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 5031
07    AA           01        01         50  F0 0074B         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                      0346  31 00751 103$:   BRW     156$                                            ; 5033
                        08   BE         05  D0 00754 104$:   MOVL    #5, @LEFT_OR_RIGHT_CVT                          ; 5042
                  00000090   8F         56  D1 00758         CMPL    STATE, #144                                     ; 5043
                                        10  12 0075F         BNEQ    105$                                            ;
                             6A    08   A9  90 00761         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 5046
      50      0A   A9        01         03  EF 00765         EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 5048
07    AA           01        01         50  F0 0076B         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                   0C   AE  D5 00771 105$:   TSTL    TURN                                            ; 5050
                                        3B  12 00774         BNEQ    109$                                            ;
                             50    0C   AC  D0 00776         MOVL    SRC_INFO, R0                                    ; 5053
                        05   A0         1F  B0 0077A         MOVW    #31, 5(R0)                                      ;
                             51    04   AC  D0 0077E         MOVL    SOURCE, R1                                      ; 5054
05    A0 00000000G 00   04   B1         61  26 00782         CVTTP   (R1), @4(R1), LIB$AB_CVTTP_U, 5(R0), @1(R0)     ; 5055
                                   01   B0     0078D                                                                 ;
                                        43  11 0078F 106$:   BRB     112$                                            ; 4690
                        08   BE         05  D0 00791 107$:   MOVL    #5, @LEFT_OR_RIGHT_CVT                          ; 5061
                  00000091   8F         56  D1 00795         CMPL    STATE, #145                                     ; 5062
                                        10  12 0079C         BNEQ    108$                                            ;
                             6A    08   A9  90 0079E         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 5065
      50      0A   A9        01         03  EF 007A2         EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 5067
07    AA           01        01         50  F0 007A8         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                   0C   AE  D5 007AE 108$:   TSTL    TURN                                            ; 5069
                                        44  12 007B1 109$:   BNEQ    115$                                            ;
                             50    0C   AC  D0 007B3         MOVL    SRC_INFO, R0                                    ; 5072
                        05   A0         1F  B0 007B7         MOVW    #31, 5(R0)                                      ;
                             51    04   AC  D0 007BB         MOVL    SOURCE, R1                                      ; 5074
                                        61  B5 007BF         TSTW    (R1)                                            ;
                                        04  12 007C1         BNEQ    110$                                            ;
                                        52  D4 007C3         CLRL    R2                                              ;
                                        05  11 007C5         BRB     111$                                            ;
                             52         61  3C 007C7 110$:   MOVZWL  (R1), R2                                        ;
                                        52  D7 007CA         DECL    R2                                              ;
01    B0      05   A0   04   B1         52  09 007CC 111$:   CVTSP   R2, @4(R1), 5(R0), @1(R0)                       ; 5075
                                      0347  31 007D4 112$:   BRW     165$                                            ; 4690
                        08   BE         05  D0 007D7 113$:   MOVL    #5, @LEFT_OR_RIGHT_CVT                          ; 5081
                  00000092   8F         56  D1 007DB         CMPL    STATE, #146                                     ; 5082
                                        10  12 007E2         BNEQ    114$                                            ;
                             6A    08   A9  90 007E4         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)                ; 5085
      50      0A   A9        01         03  EF 007E8         EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 5087
07    AA           01        01         50  F0 007EE         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)                  ;
                                   0C   AE  D5 007F4 114$:   TSTL    TURN                                            ; 5089
                                        DB  12 007F7 115$:   BNEQ    112$                                            ;
                        14   AE 00000000'   EF  9E 007F9     MOVAB   P.AKP, PACK_ZERO                                ; 5098
                             55    0C   AC  D0 00801         MOVL    SRC_INFO, R5                                    ; 5099
                        05   A5         1F  B0 00805         MOVW    #31, 5(R5)                                      ;
                             5B    04   AC  D0 00809         MOVL    SOURCE, R11                                     ; 5111
                             54         6B  3C 0080D         MOVZWL  (R11), R4                                       ;
                             54    04   AB  C0 00810         ADDL2   4(R11), R4                                      ;
                                        54  D7 00814         DECL    RT_SIGN                                         ;
```

```
                58      04  AB  D0 00816          MOVL    4(R11), LF_SIGN                         ; 5112
                52          D4 0081A              CLRL    ZERO_FLAG                               ; 5113
            41  8F          68 91 0081C          CMPB    (LF_SIGN), #65                          ; 5119
                            0C 1F 00820          BLSSU   116$
            49  8F          68 91 00822          CMPB    (LF_SIGN), #73
                            06 1A 00826          BGTRU   116$
            10  AE          01 D0 00828          MOVL    #1, SIGN_FLAG
                            65 11 0082C          BRB     122$
            4A  8F          68 91 0082E 116$:    CMPB    (LF_SIGN), #74                          ; 5123
                            0B 1F 00832          BLSSU   117$
            52  8F          68 91 00834          CMPB    (LF_SIGN), #82
                            05 1A 00838          BGTRU   117$
                    10      AE D4 0083A          CLRL    SIGN_FLAG
                            54 11 0083D          BRB     122$
            7B  8F          68 91 0083F 117$:    CMPB    (LF_SIGN), #123                         ; 5127
                            1A 12 00843          BNEQ    119$
            10  AE          01 D0 00845          MOVL    #1, SIGN_FLAG                           ; 5129
                52          01 D0 00849          MOVL    #1, ZERO_FLAG                           ; 5130
                68          30 90 0084C          MOVB    #48, (LF_SIGN)                          ; 5131
                30          64 91 0084F          CMPB    (RT_SIGN), #48                          ; 5132
                            06 12 00852          BNEQ    118$
                64      7B  8F 90 00854          MOVB    #123, (RT_SIGN)                         ; 5134
                            39 11 00858          BRB     122$
                64          10 80 0085A 118$:    ADDB2   #16, (RT_SIGN)                          ; 5136
                            34 11 0085D          BRB     122$                                    ; 5114
            7D  8F          68 91 0085F 119$:    CMPB    (LF_SIGN), #125                         ; 5141
                            19 12 00863          BNEQ    121$
                    10      AE D4 00865          CLRL    SIGN_FLAG                               ; 5143
                52          01 D0 00868          MOVL    #1, ZERO_FLAG                           ; 5144
                68          30 90 0086B          MOVB    #48, (LF_SIGN)                          ; 5145
                30          64 91 0086E          CMPB    (RT_SIGN), #48                          ; 5146
                            06 12 00871          BNEQ    120$
                64      7D  8F 90 00873          MOVB    #125, (RT_SIGN)                         ; 5148
                            1A 11 00877          BRB     122$
                64          19 80 00879 120$:    ADDB2   #25, (RT_SIGN)                          ; 5150
                            15 11 0087C          BRB     122$                                    ; 5114
                00000000'   EF 9F 0087E 121$:    PUSHAB  P.AKQ                                   ; 5153
                            01 DD 00884          PUSHL   #1
                00028362    8F DD 00886          PUSHL   #164706
    00000000G   00          03 FB 0088C          CALLS   #3, LIB$SIGNAL
                28          52 E8 00893 122$:    BLBS    ZERO_FLAG, 126$                         ; 5156
                13      10  AE E9 00896          BLBC    SIGN_FLAG, 124$                         ; 5159
                68          10 82 0089A          SUBB2   #16, (LF_SIGN)                          ; 5162
                30          64 91 0089D          CMPB    (RT_SIGN), #48                          ; 5163
                            06 12 008A0          BNEQ    123$
                64      7B  8F 90 008A2          MOVB    #123, (RT_SIGN)                         ; 5165
                            16 11 008A6          BRB     126$
                64          10 80 008A8 123$:    ADDB2   #16, (RT_SIGN)                          ; 5167
                            11 11 008AB          BRB     126$                                    ; 5159
                68          19 82 008AD 124$:    SUBB2   #25, (LF_SIGN)                          ; 5172
                30          64 91 008B0          CMPB    (RT_SIGN), #48                          ; 5173
                            06 12 008B3          BNEQ    125$
                64      7D  8F 90 008B5          MOVB    #125, (RT_SIGN)                         ; 5175
                            03 11 008B9          BRB     126$
                64          19 80 008BB 125$:    ADDB2   #25, (RT_SIGN)                          ; 5177
05  A5 00000000G   00  04  BB 6B 26 008BE 126$:  CVTTP   (R11), @4(R11), LIB$AB_CVTTP_0, 5(R5), -  ; 5183
                        01  B5     008C9          @1(R5)
```

```
                                                1E      10  AE  E9 008CB          BLBC    SIGN_FLAG, 130$                         ; 5187
                                            7B  8F          64  91 008CF          CMPB    (RT_SIGN), #123                         ; 5190
                                                            05  12 008D3          BNEQ    127$                                    ;
                                                64          30  90 008D5          MOVB    #48, (RT_SIGN)                          ; 5192
                                                            03  11 008D8          BRB     128$                                    ;
                                                64          10  82 008DA 127$:     SUBB2   #16, (RT_SIGN)                          ; 5194
                                                30          68  91 008DD 128$:     CMPB    (LF_SIGN), #48                          ; 5196
                                                            06  12 008E0          BNEQ    129$                                    ;
                                                68      7B  8F  90 008E2          MOVB    #123, (LF_SIGN)                         ; 5198
                                                            21  11 008E6          BRB     134$                                    ;
                                                68          10  80 008E8 129$:     ADDB2   #16, (LF_SIGN)                          ; 5200
                                                            1C  11 008EB          BRB     134$                                    ; 5187
                                            7D  8F          64  91 008ED 130$:     CMPB    (RT_SIGN), #125                         ; 5206
                                                            05  12 008F1          BNEQ    131$                                    ;
                                                64          30  90 008F3          MOVB    #48, (RT_SIGN)                          ; 5208
                                                            03  11 008F6          BRB     132$                                    ;
                                                64          19  82 008F8 131$:     SUBB2   #25, (RT_SIGN)                          ; 5210
                                                30          68  91 008FB 132$:     CMPB    (LF_SIGN), #48                          ; 5212
                                                            06  12 008FE          BNEQ    133$                                    ;
                                                68      7D  8F  90 00900          MOVB    #125, (LF_SIGN)                         ; 5214
                                                            03  11 00904          BRB     134$                                    ;
                                                68          19  80 00906 133$:     ADDB2   #25, (LF_SIGN)                          ; 5216
    14  BE          01      01  B5      05      A5  37  00909 134$:     CMPP4   5(R5), @T(R5), #1, @PACK_ZERO                   ; 5220
                                                54  DC 00911          MOVPSL  R4                                              ;
        54          54          02              02  EF 00913          EXTZV   #2, #2, R4, R4                                  ;
                                                54  D7 00918          DECL    R4                                              ;
                                                68  12 0091A          BNEQ    139$                                            ;
                                            50      05  A5  3C 0091C          MOVZWL  5(R5), R0                               ; 5222
                                            50          02  C6 00920          DIVL2   #2, R0
                                51  00000000G  00      04  AB  C1 00923          ADDL3   4(R11), LIB$AB_CVTTP_0, R1              ; 5223
01  B540                        04              00          61  F0 0092C          INSV    (R1), #0, #4, @1(R5)[R0]                ; 5222
                                                4F  11 00933          BRB     139$                                            ; 4690
                                        08      BE          05  D0 00935 135$:     MOVL    #5, @LEFT_OR_RIGHT_CVT                  ; 5229
                                00000093        8F          56  D1 00939          CMPL    STATE, #147                             ; 5230
                                                            10  12 00940          BNEQ    136$                                    ;
                                                6A      08  A9  90 00942          MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 5233
        50          0A  A9                      01          03  EF 00946          EXTZV   #3, #1, T0(SRC_OR_DST), R0              ; 5235
07  AA              01                          01          50  F0 0094C          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)          ;
                                                0C      AE  D5 00952 136$:     TSTL    TURN                                    ; 5237
                                                            6A  12 00955          BNEQ    142$                                    ;
                                            50      04  AC  D0 00957          MOVL    SOURCE, R0                              ; 5244
                                                            60  B5 0095B          TSTW    (R0)                                    ;
                                                            04  12 0095D          BNEQ    137$                                    ;
                                                            5B  D4 0095F          CLRL    SOU_LEN                                 ;
                                                            05  11 00961          BRB     138$                                    ;
                                                5B          60  3C 00963 137$:     MOVZWL  (R0), SOU_LEN                           ;
                                                            5B  D7 00966          DECL    SOU_LEN                                 ;
                                                20      AE  04  B0 04B 90 00968 138$:     MOVB    @4(R0)[SOU_LEN], TEMP_BUF       ; 5246
                                    21  AE          04      B0  5B  28 0096E          MOVC3   SOU_LEN, @4(R0), TEMP_BUF+1             ; 5247
                                                            50  0C  AC  D0 00974          MOVL    SRC_INFO, R0                    ; 5248
                                                05      A0          1F  B0 00978          MOVW    #31, 5(R0)                      ;
01  B0          05  A0              20      AE          5B  09 0097C          CVTSP   SOU_LEN, TEMP_BUF, 5(R0), @1(R0)        ; 5249
                                                            78  11 00984 139$:     BRB     145$                                    ; 4690
                                                08      BE          05  D0 00986 140$:     MOVL    #5, @LEFT_OR_RIGHT_CVT          ; 5255
                                        00000094        8F          56  D1 0098A          CMPL    STATE, #148                     ; 5256
                                                            10  12 00991          BNEQ    141$                                    ;
                                                6A      08  A9  90 00993          MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 5259
```

```
                                                                03  EF  00997             EXTZV   #3, #1, 10(SRC_OR_DST), R0                      ; 5261
      50      0A   A9                    01                     07  AA  01                INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                                0C  AE  D5  009A3 141$:     TSTL    TURN                                        ; 5263
                                                                    78  12  009A6         BNEQ    148$
                                                                0C  AC  D0  009A8         MOVL    SRC_INFO, R0                                ; 5266
                                                                    1F  B0  009AC         MOVW    #31, 5(R0)
                                                                04  AC  D0  009B0         MOVL    SOURCE, R1                                  ; 5267
      05   A0 00000000G   00          04  B1                     61  26  009B4         CVTTP   (R1), @4(R1), LIB$AB_CVTTP_0, 5(R0), @1(R0) ; 5268
                                                                01  B0      009BF
                                                                    7F  11  009C1 142$:     BRB     150$                                        ; 4690
                                                                    05  D0  009C3 143$:     MOVL    #5, @LEFT_OR_RIGHT_CVT                      ; 5274
                                  00000095  8F                      56  D1  009C7         CMPL    STATE, #149                                 ; 5275
                                                                    10  12  009CE         BNEQ    144$
                                                                08  A9  90  009D0         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)            ; 5278
      50      0A   A9                    01                     03  EF  009D4         EXTZV   #3, #7, 7(SRC_OR_DST), R0                   ; 5280
      07  AA       01                    01                     50  F0  009DA         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                                0C  AE  D5  009E0 144$:     TSTL    TURN                                        ; 5282
                                                                    5D  12  009E3         BNEQ    150$
                                                                0C  AC  D0  009E5         MOVL    SRC_INFO, R0                                ; 5285
                                                                    1F  B0  009E9         MOVW    #31, 5(R0)
                                                                04  AC  D0  009ED         MOVL    SOURCE, R1                                  ; 5286
      05   A0 00000000G   00          04  B1                     61  26  009F1         CVTTP   (R1), @4(R1), LIB$AB_CVTTP_Z, 5(R0), @1(R0) ; 5287
                                                                01  B0      009FC
                                                                    42  11  009FE 145$:     BRB     150$                                        ; 4690
                                                                    05  D0  00A00 146$:     MOVL    #5, @LEFT_OR_RIGHT_CVT                      ; 5293
                                  00000096  8F                      56  D1  00A04         CMPL    STATE, #150                                 ; 5294
                                                                    10  12  00A0B         BNEQ    147$
                                                                08  A9  90  00A0D         MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)            ; 5297
      50      0A   A9                    01                     03  EF  00A11         EXTZV   #3, #7, 7(SRC_OR_DST), R0                   ; 5299
      07  AA       01                    01                     50  F0  00A17         INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)
                                                                0C  AE  D5  00A1D 147$:     TSTL    TURN                                        ; 5301
                                                                    7B  12  00A20 148$:     BNEQ    157$
                                                                04  AC  D0  00A22         MOVL    SOURCE, R0                                  ; 5304
      20  AE           1F            04  B0                     60  08  00A26         CVTPS   (R0), @4(R0), #31, TEMP_BUF
                                                                0C  AC  D0  00A2D         MOVL    SRC_INFO, R4                                ; 5305
      01  B4           1F            20  AE                     1F  09  00A31         CVTSP   #31, TEMP_BUF, #31, @1(R4)
                                     05  A4                     1F  B0  00A38         MOVW    #31, 5(R4)                                  ; 5306
                                                                    6E  11  00A3C         BRB     158$                                        ; 4690
                                     0B  BE                     06  D0  00A3E 149$:     MOVL    #6, @LEFT_OR_RIGHT_CVT                      ; 5311
                                                                    68  11  00A42 150$:     BRB     158$
                                     0B  BE                     06  D0  00A44 151$:     MOVL    #6, @LEFT_OR_RIGHT_CVT                      ; 5315
                                     05  AA                     69  B0  00A48         MOVW    (SRC_OR_DST), 5(SRC_OR_DST_INFO)            ; 5316
                                                                    4C  11  00A4C         BRB     156$                                        ; 5317
                                     0B  BE                     06  D0  00A4E 152$:     MOVL    #6, @LEFT_OR_RIGHT_CVT                      ; 5326
                                  0000FFFF  8F              0C  A9  D1  00A52             CMPL    12(SRC_OR_DST), #65535                      ; 5327
                                                                23  14  00A5A             BGTR    154$
                                     01                     0B  A9  91  00A5C             CMPB    11(SRC_OR_DST), #1
                                                                1D  12  00A60             BNEQ    154$
                                     01                         69  B1  00A62             CMPW    (SRC_OR_DST), #1                            ; 5328
                                                                18  12  00A65             BNEQ    154$
                                  000000AE  8F                      56  D1  00A67             CMPL    STATE, #174                                 ; 5331
                                                                    09  13  00A6E             BEQL    153$
                                  000000BA  8F                      56  D1  00A70             CMPL    STATE, #186
                                                                    0C  12  00A77             BNEQ    155$
                                     01                     14  A9  D1  00A79 153$:         CMPL    20(SRC_OR_DST), #1                          ; 5334
                                                                06  13  00A7D             BEQL    155$
                                     50                         07  CE  00A7F 154$:         MNEGL   #7, STATUS
```

```
                                                00A8  31 00A82          BRW     167$                                    ;
                                        6A      08 A9  90 00A85 155$:    MOVB    8(SRC_OR_DST), (SRC_OR_DST_INFO)        ; 5336
        50      0A      A9                      01 03  EF 00A89          EXTZV   #3, #1, 10(SRC_OR_DST), R0              ; 5337
  07    AA              01                      01 50  F0 00A8F          INSV    R0, #1, #1, 7(SRC_OR_DST_INFO)          ;
                                05      AA      0C A9  B0 00A95          MOVW    12(SRC_OR_DST), 5(SRC_OR_DST_INFO)      ; 5338
                                                0C AE  D5 00A9A 156$:    TSTL    TURN                                    ; 5339
                                                7F 12  00A9D 157$:       BNEQ    165$                                    ;
                                        51      0C AC  D0 00A9F          MOVL    SRC_INFO, R1                            ; 5342
                                        50      04 AC  D0 00AA3          MOVL    SOURCE, R0                              ;
                                01      A1      04 A0  D0 00AA7          MOVL    4(R0), 1(R1)                            ;
                                                70 11  00AAC 158$:       BRB     165$                                    ; 4690
                                08      BE          06 D0  00AAE 159$:   MOVL    #6, @LEFT_OR_RIGHT_CVT                  ; 5348
                                                0C AE  D5 00AB2          TSTL    TURN                                    ; 5349
                                                5E 12  00AB5             BNEQ    164$                                    ;
                                        50      0C AC  D0 00AB7          MOVL    SRC_INFO, R0                            ; 5352
                                        51      04 AC  D0 00ABB          MOVL    SOURCE, R1                              ;
                        01      A0      04 A1  02 C1  00ABF              ADDL3   #2, 4(R1), 1(R0)                        ;
                                05      A0      04 B1  B0 00AC5          MOVW    @4(R1), 5(R0)                           ; 5353
                                                52 11  00ACA             BRB     165$                                    ; 5349
                                08      BE          06 D0  00ACC 160$:   MOVL    #6, @LEFT_OR_RIGHT_CVT                  ; 5361
                                                0C AE  D5 00AD0          TSTL    TURN                                    ; 5362
                                                40 12  00AD3             BNEQ    164$                                    ;
                                        50      0C AC  D0 00AD5          MOVL    SRC_INFO, R0                            ; 5365
                                        51      04 AC  D0 00AD9          MOVL    SOURCE, R1                              ;
                        01      A0      04 A1  01 C1  00ADD              ADDL3   #1, 4(R1), 1(R0)                        ;
                                05      A0      04 B1  9B 00AE3          MOVZBW  @4(R1), 5(R0)                           ; 5366
                                                34 11  00AE8             BRB     165$                                    ; 5362
                                08      BE          06 D0  00AEA 161$:   MOVL    #6, @LEFT_OR_RIGHT_CVT                  ; 5374
                                                0C AE  D5 00AEE          TSTL    TURN                                    ; 5375
                                                22 12  00AF1             BNEQ    164$                                    ;
                                                52 D4  00AF3             CLRL    COUNT                                   ; 5381
                                        51      04 AC  D0 00AF5          MOVL    SOURCE, R1                              ; 5382
                                        50      04 A1  D0 00AF9          MOVL    4(R1), SRC_PTR                          ;
                                                6240 95  00AFD 162$:     TSTB    (COUNT)[SRC_PTR]                        ; 5383
                                                04 13  00B00             BEQL    163$                                    ;
                                                52 D6  00B02             INCL    COUNT                                   ; 5384
                                                F7 11  00B04             BRB     162$                                    ;
                                        50      0C AC  D0 00B06 163$:    MOVL    SRC_INFO, R0                            ; 5385
                                05      A0          52 B0  00B0A         MOVW    COUNT, 5(R0)                            ;
                                01      A0      04 A1  D0 00B0E          MOVL    4(R1), 1(R0)                            ; 5386
                                                09 11  00B13             BRB     165$                                    ; 5375
                                        50      10 AC  D0 00B15 164$:    MOVL    DST_INFO, R0                            ; 5389
                                05      A0      08 BC  B0 00B19          MOVW    @DESTINATION, 5(R0)                     ;
                                                0C AE  D6 00B1E 165$:    INCL    TURN                                    ; 4597
                                        03      0C AE  D1 00B21          CMPL    TURN, #3                                ;
                                                03 1A  00B25             BGTRU   166$                                    ;
                                                F4DF 31  00B27           BRW     1$                                      ;
                                        50          01 CE  00B2A 166$:   MNEGL   #1, STATUS                              ;
                        51      18      AE      06 C5  00B2D 167$:       MULL3   #6, LEFT_CVT, R1                        ; 5404
                                        51      1C AE  C0 00B32          ADDL2   RIGHT_CVT, R1                           ;
                                14      BC      FA A1  9E 00B36          MOVAB   -6(R1), @CVT_PATH                       ;
                                                    04 00B3B             RET                                             ; 5406

; Routine Size:  2876 bytes,    Routine Base:  DBG$CODE + 347F

; 5304          5407  1
```

```
; 5305          5408  1 END                       ! End of module DBGCVTDX.
; 5306          5409  0 ELUDOM


                                                    .EXTRN  LIB$SIGNAL, SYS$UNWIND

;                               PSECT SUMMARY

;       Name                    Bytes                   Attributes

;  DBG$OWN                       208  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,  PIC,ALIGN(2)
;  DBG$PLIT                     6740  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(0)
;  DBG$CODE                    16315  NOVEC,NOWRT,  RD ,  EXE,  SHR,  LCL,  REL,  CON,  PIC,ALIGN(0)


;                           Library Statistics

;                                    --------  Symbols  --------      Pages       Processing
;       File                         Total     Loaded   Percent      Mapped      Time

;  _$255$DUA28:[SYSLIB]LIB.L32;1      18619         59         0        1000        00:01.8
;  _$255$DUA28:[DEBUG.OBJ]STRUCDEF.L32;1
;                                        32          0         0           7        00:00.1
;  _$255$DUA28:[DEBUG.OBJ]DBGLIB.L32;1
;                                      1545        101         6          97        00:02.0
;  _$255$DUA28:[DEBUG.OBJ]DSTRECRDS.L32;1
;                                       418          6         1          31        00:00.3
;  _$255$DUA28:[DEBUG.OBJ]DBGMSG.L32;1
;                                       386         18         4          22        00:00.3


; Information:  2
; Warnings:     0
; Errors:       0


;                                   COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:DBGCVTDX/OBJ=OBJ$:DBGCVTDX MSRC$:DBGCVTDX/UPDATE=(ENH$:DBGCVTDX)

; Size:         16315 code + 6948 data bytes
; Run Time:       05:24.7
; Elapsed Time:   17:37.9
; Lines/CPU Min:    999
; Lexemes/CPU-Min: 14517
; Memory Used:  2755 pages
; Compilation Complete
```

DBGCALL
LIS

DBGCVTDX
LIS

DBGDPC
LIS

DBGCVTMAC
LIS

DBGDEFINE
LIS